

Modeling of Ion Thruster Discharge Chamber Using 3D Particle-In-Cell Monte-Carlo-Collision Method

Binh Phuoc Tran

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Aerospace Engineering

Dr. Joseph J. Wang, Committee Chair
Dr. Robert W. Walters, Committee Member
Dr. Wayne A. Scales, Committee Member

June 2, 2005
Blacksburg, Virginia

Keywords: 3D PIC MCC Simulation, Ion Thruster Discharge Chamber
Copyright 2005, Binh Phuoc Tran

Modeling of Ion Thruster Discharge Chamber Using 3D Particle-In-Cell
Monte-Carlo-Collision Method

Binh Phuoc Tran

Abstract

This thesis is aimed toward developing a method to simulate ion thruster discharge chambers in a full three dimensional environment and to study the effect of discharge chamber size on ion thruster performance. The study focuses solely on *ring-cusped* thrusters that make use of *Xenon* for propellant and discharge cathode assembly for mean of propellant ionization. Commercial software is used in both the setup and analysis phases. Numerical simulation is handled by 3D Particle-In-Cell Monte-Carlo-Collision method. Simulation results are analyzed and compared with other works. It is concluded that the simulation methodology is validated and can be used to simulate different cases. Therefore, different simulation cases of varying chamber sizes are done and the results are used to develop a performance curve. This plot suggests that the most efficient case is the 30 cm thruster. The result further validates the simulation process since the operating parameters used for all of the cases are taken from a 30 cm thruster experiment. One of the obvious applications for such a simulation process is to determine a set of the most efficient operating parameters for a certain size thruster before actual fabrication and laboratory testing.

Acknowledgments

I would like to thank my advisor Dr. Wang for without him, this project would not have even begun. He taught me most of what I know about the subject matters of plasma dynamics, propulsion, and simulation techniques. I would like to thank him for all the patience he has shown me throughout the years as I struggled to understand the above concepts that were so alien to me. Thank you for giving me another shot at working on numerical simulation after taking a year off research at the beginning of my graduate career.

I would like to thank Lubos Brieda for all of his help and advice throughout this thesis project. Also, for taking his time out of his own thesis work to answer the numerous coding and running errors that I ran into during the code development stage.

I would like to thank our department system administrator Luke Scharf. All those computer talks when my mind did not want to think simulation anymore were very educational and interesting. All his timely support made the project go by smoother. Thank you for that big save near the end of the road when my laptop died on me.

I would like to thank Dr. Robert W. Walters and Dr. Wayne A. Scales for serving on my graduate committee. I know this means taking time off of their already busy schedules to help me.

I would like to thank the Aerospace and Graduate departments staff for all their all their advice and support to get everything filled out and signed in order to graduate.

Contents

1	Introduction	1
1.1	Ring-Cusped Ion Thruster	1
1.2	Ionization mechanism	4
1.3	Magnetic Field	5
1.3.1	Magnetic Mirror	6
1.3.2	Electron Lost to Magnetic Cusps	8
1.4	Thruster Performance Model	8
1.5	Thesis Topic and Approach	11
2	Simulation model	12
2.1	Particle-In-Cell Code	12
2.2	Monte Carlo Collision	15
2.3	Mesh Generation	18
2.4	Magnetic field generation	19
3	Discharge chamber simulation	25
3.1	Model Setup	25
3.2	Scaling	27
3.3	Results	29
3.3.1	Analysis and validation of simulation results (30 cm case)	29
3.3.2	20 cm case	37
3.3.3	25 cm case	38
3.3.4	35 cm case	42
3.3.5	40 cm case	42
4	Application in the Performance model	48
4.1	Performance curve development	48
4.2	Electron loss behavior	53
5	Conclusions and Future Work	61

List of Tables

1.1	Xenon-electron ionization cross-sectional area	4
3.1	Input parameters for simulation cases	29
4.1	Surfaces flux data	52
4.2	Ion flux and average beam ion energy cost data	52
4.3	Electron loss flux	54

List of Figures

1.1	Ion thruster thrust (NASA Picture)	2
1.2	A typical ring-cusped ion thruster	3
1.3	Ionization cross section	5
1.4	Magnetic mirror	6
2.1	Checked formation of PIC code	13
2.2	Force weighting	14
2.3	3D meshed discharge chamber	18
2.4	Step 1 of shape and magnet configuration	20
2.5	Electrons trapped in the back of chamber	21
2.6	Path of electrons trapped in back section of chamber	21
2.7	Step 2 of shape configuration	22
2.8	Original magnetic field strength	23
2.9	Final magnets configuration	23
2.10	Magnetic field contour	24
3.1	Electric potential before simulation	30
3.2	Magnetic field lines distribution	31
3.3	Electron motion showing gyro radii	32
3.4	Particles population of 30 cm thruster after 1000 time steps	32
3.5	Ion population plot	33
3.6	Ion population of a 30 cm thruster for the first 50 time steps	34
3.7	Time step size history of the 30 cm thruster simulation	35
3.8	Electron contour plot for 30 cm thruster	35
3.9	Electron distribution plot for 30 cm thruster	36
3.10	Ion density plot for 30 cm thruster	37
3.11	20 and 30 cm thruster magnetic field strength	38
3.12	Electron distribution for the 20 cm thruster	39
3.13	Ion population for the 20 cm case	39
3.14	Ion distribution for the 20 cm thruster	40
3.15	25 and 30 cm thruster magnetic field strength	40
3.16	Electron distribution for the 25 cm thruster case	41
3.17	Ion population for the 25 cm case	41
3.18	Ion distribution for the 25 cm case	42

3.19	Magnetic field for 35 cm and 30 cm thrusters	43
3.20	Electron distribution for 35 cm case	43
3.21	Ion population for the 35 cm case	44
3.22	Ion distribution for 35 cm case	44
3.23	Magnetic field for 40 cm and 30 cm cases	45
3.24	Electron distribution for 40 cm thruster	45
3.25	Ion population for the 40 cm case	46
3.26	Ion distribution for 40 cm case	46
3.27	Ionization rate for the discharge chambers	47
4.1	Ion-surfaces flux for 20 cm thruster	49
4.2	Ion-surfaces flux for 25 cm thruster	50
4.3	Ion-surfaces flux for 30 cm thruster	50
4.4	Ion-surfaces flux for 35 cm thruster	51
4.5	Ion-surfaces flux for 40 cm thruster	51
4.6	Average beam ion energy cost	53
4.7	Electron lost for 20 cm case	54
4.8	Electron lost for 25 cm case	55
4.9	Electron lost for 30 cm case	55
4.10	Electron lost for 35 cm case	56
4.11	Electron lost for 40 cm case	56
4.12	Electron loss to grid and wall behavior	57
4.13	Ratio of electron flux to wall over grid	57
4.14	Magnet configuration and strength plot for 4 magnets	59
4.15	Electron distribution for 4 magnets case	59
4.16	Electron flux to wall for 3 and 4 rings configuration	60
4.17	Electron flux to grid for 3 and 4 rings configuration	60
A-1	Sample neutral output, top	65
A-2	Sample neutral output, bottom	66
A-3	Sample output of TECPLOT format file, top	69
A-4	Sample output of TECPLOT format file, bottom	70
B-1	Sample of B field 2d file	71
B-2	Sample 3D B field file	75

Chapter 1

Introduction

There are more and more satellites occupying the Earth's orbits as the need for their constant presence rises. Electric propulsion systems are good options for station keeping and orbit maintenance. The reason is that thrust generated is of high specific impulse, often around an order of magnitude more than conventional chemical thrusters. Due to this high specific impulse nature, less propellant mass is required. Electric propulsion systems accelerate propellant atoms through an electric field, a magnetic field, or both. The propulsion system focused in this thesis is the *ring-cusped ion propulsion* system. This system uses a static electric field for thrust generation. Thrust generated by ion thrusters, as shown in figure 1.1 [6], is done by electrostatically accelerating ions through grids located at the end of the discharge chamber.

1.1 Ring-Cusped Ion Thruster

A diagram of a typical ring-cusped ion thruster is shown in figure 1.2. The discharge chamber is where the ionization occurs. Three magnetic rings are represented by the six blocks. At the end of the discharge chamber, there are two grids. The one closest to the discharge chamber is called a screen grid and the one after that is called an acceleration grid. Grid potential at the screen grid is close to the value of plasma potential and the potential for the acceleration grid is a relatively small negative value. As shown, the screen grid potential is $+1070V$ and acceleration grid potential is $-200V$. The second cathode external to the thruster is used to neutralize the exit plasma plume.

The ion propulsion system under study uses *Xenon* as propellant. Xenon is injected at the central axis of the discharge chamber at the location of the cathode. It then moves around inside the chamber until ionization occurs with the primary electrons emitted from the discharge chamber cathode. As

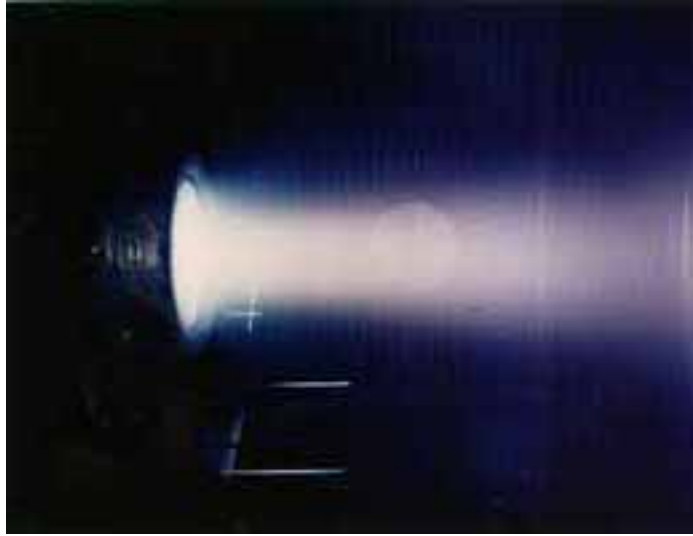


Figure 1.1: Ion thruster thrust (NASA Picture)

particles are moving around, they may run into the wall surfaces and lose charge. In the case of electrons, they are absorbed completely by the wall surfaces. Ions, on the other hand, only lose their charge and get reflected back as neutral particles.

Discharge chamber particles move around according to their thermal velocities and collisional processes and if applicable, ionization. As ionization occurs, an environment of quasi-neutral plasma is created. They will either get reflected back or absorbed accordingly. Electrons are lost to all surfaces they come in contact with due to their light weight and charged nature whereas ions and neutrals have different fates. Since neutrals have no charge, they only move under thermal and collision induced forces. They simply bounce off the wall when they come into contact with it. This will happen until they happen to wander into the holes in the grids and flow out of the thruster. On the other hand, ions have charge and will interact with the charged chamber wall and grids. Along the wall and grids there exist a plasma sheath. The sheath is a limit of how far the charged surface can reach into the plasma before it is shielded out by the plasma. In order for ions to reach the wall, they must have enough kinetic energy to overcome the force induced by the charge of the surfaces. Since the grids are filled with holes making about seventy percent of the surfaces hollow, a number of ions will both have enough kinetic energy and be able to travel through the grid slots. These are the ions that are used to create thrust. The accelerator grid behind the screen grid possesses a negative voltage therefore creating an electrostatic field with a gradient pointing in the axial direction toward

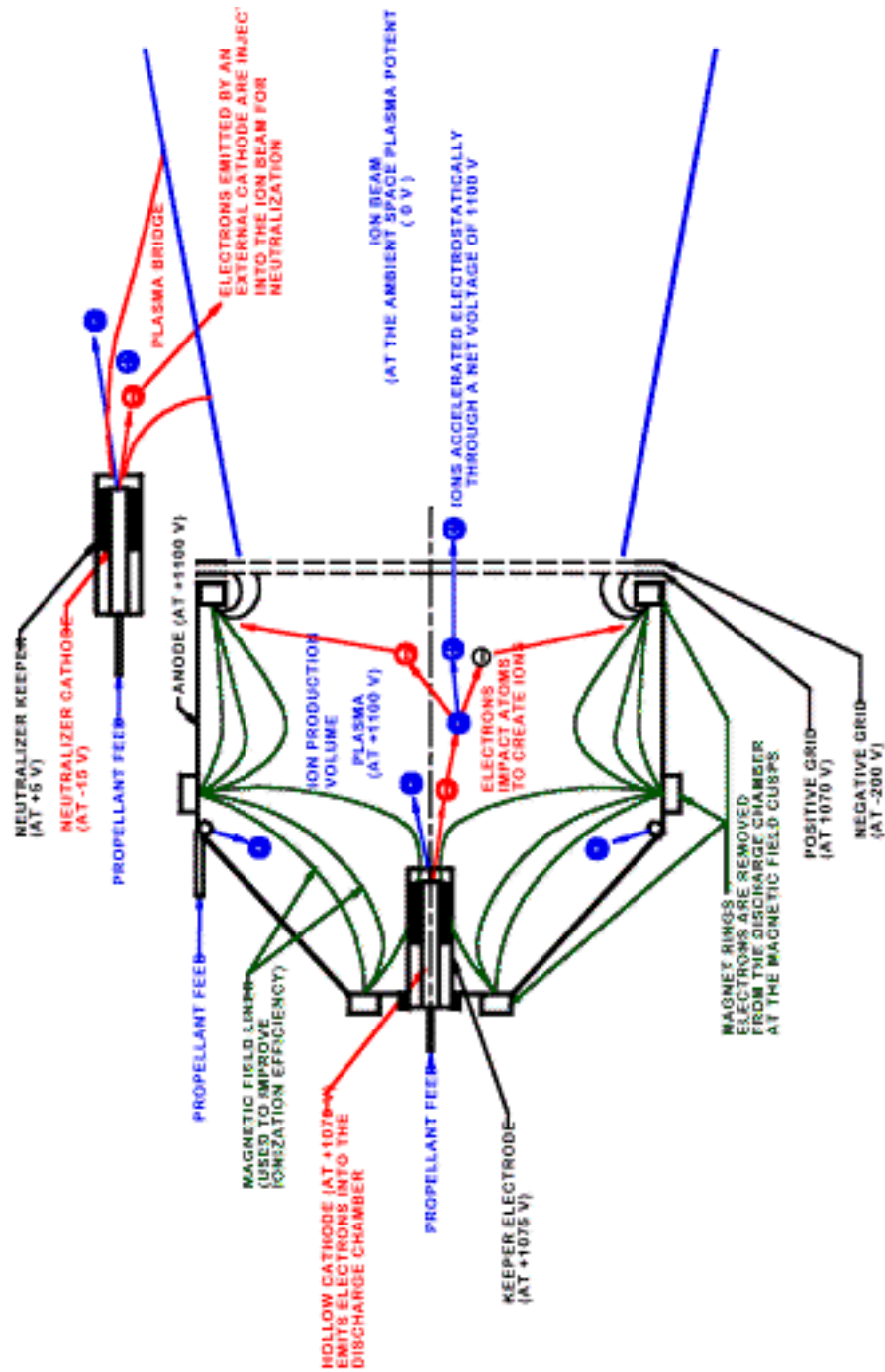


Figure 1.2: A schematic of an ring-cusped ion thruster.

$\epsilon(eV)$	$q_i (\text{\AA}^2)$
10	0
15	0.906
20	2.28
30	3.85
50	4.85
100	5.57
200	4.85
300	4.06
500	3.05
1000	1.94

Table 1.1: Xenon-electron ionization cross-sectional area

the outside of the thruster. This electric field accelerates the ions through the grids, producing the thrust.

When charged particles move far enough downstream toward the thruster grids, they will encounter a plasma sheath generated by the potential of the screen grid. The particles that have thermal energy that is larger than the required energy to cross the sheath, the Bohm's condition, will be extracted from the chamber and accelerate out to create thrust. The exhaust plume consists of leaked neutrals and accelerated ions. If the exhaust plume is not neutralized, it also generates a virtual anode aft of the exit plane. This virtual anode is a concentration of positive charges that prevents more ions from accelerating away from the thruster. A cathode is placed outside of the thruster where electrons are emitted into the exhaust plume creating a neutral plasma plume.

1.2 Ionization mechanism

Ionization of gas can happen from four main processes. The primary ionization process that happens inside an ion thruster discharge chamber is inelastic collision due to bombardment of electrons emitted from the discharge cathode. Electrons that are injected into the discharge chamber from discharge cathode are called primary electrons. These particles are the most energetic in the chamber. After ionization, there are chances that ions will recombine with some electrons to become neutrals once again. Electron-ion recombination is not included in the current model because the recombination length greatly exceeds that of the ionization mean free path. A data set of the ionization cross section of colliding species is obtained from [8]. This data is shown in table 1.1 and figure 1.3. From this data, the electron-

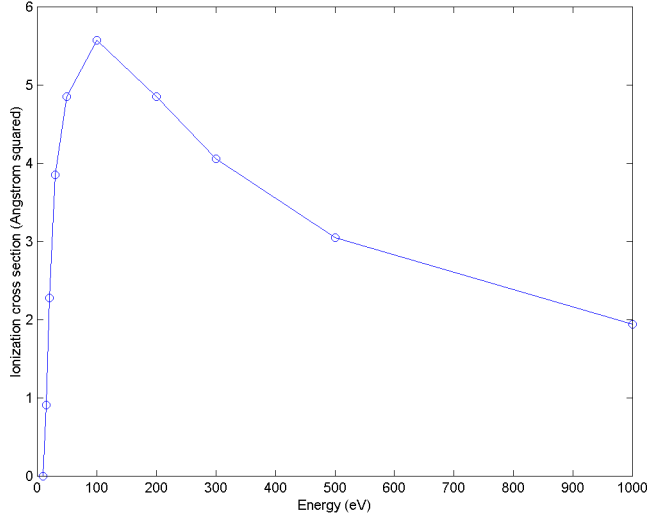


Figure 1.3: Plot of ionization cross section for electron-xenon collisions.

neutral collision rate can be calculated by a *collision frequency*

$$\nu = \sigma v n \quad (1.1)$$

where σ is the cross-sectional area, v is the relative velocities between electrons and neutrals, and n is the number density of the neutral. This is used in the Monte Carlo Collision calculation.

1.3 Magnetic Field

The discharge chamber magnetic field is generated from three ring magnets. This field is intended to both contain the charged particles and to maximize ionization probability [2]. Containment is done through particle motion about magnetic field lines. When a charged particle has a velocity component perpendicular to the magnetic field, this generates a force perpendicular of both the field and velocity.

$$ma = q(v \times B) \quad (1.2)$$

This in effect causes the particle to gyrate about the field line. Positively charged particles gyrate in the opposite direction of negatively charged particles. Particles gyration distance from the field line, or *gyro radius* or *Larmor radius*, is dependent on their respective mass.

$$\rho_L = \frac{mv_{\perp}}{|q| B} \quad (1.3)$$

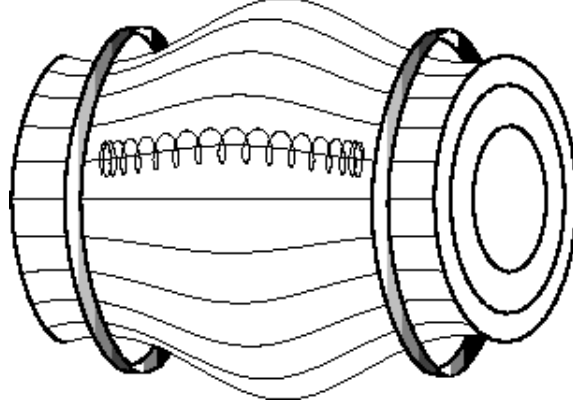


Figure 1.4: A typical schematic of a magnetic mirror where the charged particles are trapped in the low field volume.

1.3.1 Magnetic Mirror

Let us use a magnetic mirror to illustrate the confinement of charged particles by a magnetic field, as shown in figure 1.4 [5]. In the presence of a non-uniform magnetic field, the space where the density of the field lines changes, the charged particles see an induced force opposite of this density gradient. This force slows the particles down and eventually traps them in the field space. The concept is demonstrated mathematically in the following. Take the Lorentz Force Law

$$\vec{F} = q (\vec{v} \times \vec{B}) \quad (1.4)$$

decomposing this into three components in the cylindrical coordinate system with magnetic direction parallel with the z direction will result in

$$F_r = q (v_\theta B_z - v_z B_\theta) \quad (1.5)$$

$$F_\theta = q (-v_r B_z + v_z B_r) \quad (1.6)$$

$$F_z = q (v_r B_\theta - v_\theta B_r) \quad (1.7)$$

Due to the direction of the magnetic field, there is no field component in the θ direction therefore eliminating the parts with B_θ . This leads to

$$F_r = q v_\theta B_z \quad (1.8)$$

$$F_\theta = q (-v_r B_z + v_z B_r) \quad (1.9)$$

$$F_z = -q v_\theta B_r \quad (1.10)$$

Here, F_r and the first term of F_θ represent the gyration motion of the charged particle about B_z . The force along the z direction, F_z , is the force

responsible for particle confinement. When there is a change in field density that is parallel to z , $\nabla B \parallel B_z$, B_r has value and can be derive as follows

$$rB_r = - \int \frac{r\partial B_z}{\partial z} dr \quad (1.11)$$

If the variation of B_z on z is small, then

$$rB_r \approx - \frac{\partial B_z}{\partial z} \int r dr = -\frac{1}{2}r^2 \left(\frac{\partial B_z}{\partial z} \right) \quad (1.12)$$

This leads to

$$B_r \approx -\frac{1}{2}r \left(\frac{\partial B_z}{\partial z} \right) \quad (1.13)$$

Transforming F_z to

$$F_z = -qv_\theta B_r \approx \frac{1}{2}qv_\theta r \left(\frac{\partial B_z}{\partial z} \right) \quad (1.14)$$

Here, v_θ is redefined as the velocity component perpendicular to the field direction, $v_\perp$, and r as the Larmor radius, r_L . Also, a new variable is defined as the *gyration frequency*

$$\omega_c = \frac{v_\perp}{r_L} \quad (1.15)$$

to get

$$F_z = -\frac{1}{2}q \frac{v_\perp^2}{\omega_c} \frac{\partial B_z}{\partial z} \quad (1.16)$$

and since the magnitude of the magnetic field can be defined as

$$B = \frac{\omega_c m}{q} \quad (1.17)$$

Then F_z can be reduced to

$$F_z = -\frac{1}{2} \frac{mv_\perp^2}{B} \frac{\partial B_z}{\partial z} \quad (1.18)$$

It is recognized that the mass of the particle, the perpendicular speed, and the magnetic field magnitude can be assumed to be constant. This defines another variable called *magnetic moment*

$$\mu = \frac{1}{2} \frac{mv_\perp^2}{B} \quad (1.19)$$

This leads to the final version of the force in the z direction as

$$F_z = -\mu \frac{\partial B_z}{\partial z} \quad (1.20)$$

or in a more general form that defines the force in the direction of the magnetic field as

$$F_{\parallel} = -\mu \nabla_{\parallel} \vec{B} \quad (1.21)$$

This force slows down the particles to a point of the gradient where the particles lose all their momentum and start moving backward. If there is also a gradient that exists at the other direction that serve the opposite purpose, then a magnetic mirror is created. This is the same concept that the magnetic field inside discharge chambers use to confine charged particles.

1.3.2 Electron Lost to Magnetic Cusps

One of the challenges for an ion thruster designer is how to keep the primary electrons in the chamber long enough so they can ionize the propellant. A major loss of these primary electrons is when they hit the chamber at magnetic cusps. Magnetic cusps are regions of concentrated magnetic field, typically the sources of the magnetic fields. These sources either come from permanent magnets or electromagnets. The reason why most of the electrons are lost there is because of magnetic mirror confinement. Most electrons will simply travel back and forth in the mirror volume if their energies are low enough and the magnetic field gradient are suitable for complete containment. However, some electrons have a sufficient residual velocity at the cusps and proceed to ram into the chamber wall losing their energies and charges. Energy that was expended to create this electron is now lost from the thrust generation process. Wall absorption of electrons also means that there are fewer electrons in the chamber to ionize the propellant. Therefore, this phenomenon serves both to increase power consumption and to lower efficiency of the thruster. Due to this, care must be taken to decide upon the magnetic configuration in the discharge chamber.

1.4 Thruster Performance Model

The performance model used in this thesis is based on the Brophy model [2]. This simple model, which has been experimentally verified [3], is ideal due to the fact that only thruster geometry and propellant usage information are needed to develop a performance model of the thruster. This section will describe the model in more detail and show how it will be used in this thesis. The main goal of this model is to find the average energy cost to produce a beam ion. This beam ion is the particle that is actually used in making thrust. The model focuses mainly on the energy conservation of particle loss. Loss of particles can range from primary electron loss to cusps to ion lost to grids. We will start with a very general description of an *energy cost*

per beam ion as

$$\epsilon_B = (J_D - J_B) \frac{V_D}{J_B} \quad (1.22)$$

with unit as $eV/\text{beam ion}$ and where J_D is the discharge current, J_B is the beam current, and V_D is the discharge voltage. Here, J_D is subtracted from J_D to account for the fact that energy usage to accelerate ions to create thrust is not included in the beam ion cost analysis. Also define the *average plasma ion energy cost*, or the average energy cost to produce a plasma ion in the discharge chamber, as

$$\epsilon_P = (J_D - (J_C + J_B)) \frac{V_D}{J_P} \quad (1.23)$$

where J_C is the ion current to the cathode potential surfaces and J_P is the ion production current, both measured in amps (A). Here, J_C is also taken out so that the energy loss to accelerating ions to the wall is not included in the energy cost of making ions. Rearranging 1.23 to get

$$\epsilon_P = \frac{J_D - J_B}{J_B} V_D \frac{J_B}{J_P} - \frac{J_C}{J_P} V_D \quad (1.24)$$

from 1.24 we can define two fractions

$$f_B = \frac{J_B}{J_P} \quad (1.25)$$

and

$$f_C = \frac{J_C}{J_P} \quad (1.26)$$

where J_B is the fraction of produced ions that go into the thrust beam and J_C is the fraction of ions that go into the chamber cathode surfaces. Now, put 1.22 into 1.24 gives

$$\epsilon_P = \epsilon_B f_B - f_C V_D \quad (1.27)$$

Solving for ϵ_B one obtains

$$\epsilon_B = \frac{\epsilon_P}{f_B} + \frac{f_C V_D}{f_B} \quad (1.28)$$

We now have an equation for the beam ion energy cost in terms of plasma ion energy cost (ϵ_P), the extracted ion fraction (f_B), the cathode loss fraction (f_C), and the discharge voltage (V_D). Here, the first term on the right hand side represents the energy loss to creating the plasma ions and only extracting a portion of that to produce thrust. The second term represents the energy loss in accelerating ions to cathode surfaces. Equation 1.28 is the main equation to be used. The next paragraph will discuss more about the mechanism of the energy required to create ions in the discharge

chamber.

The plasma ion energy cost variable ϵ_P encompasses all aspects of energies that goes into an ion less the energy to accelerate it through the grids. This process starts with the energy required to create and injected primary electrons into the chamber. There are several actions that these primary electrons can take, all of which will involve them giving up their energies.

- Collision with neutrals resulting in ionization creating ions and secondary electrons
- Collision with neutrals resulting in excitation
- Primary electron lost to cusps or chamber surfaces
- Secondary electron lost to chamber surfaces

Any actions other than ionization of neutrals can be considered energy wasted since these processes do not contribute to the creation of thrust. The entire process is described in mathematical details in [2] resulting in

$$\epsilon_P = \frac{\epsilon_P^*}{1 - \exp[-C_0 \dot{m} (1 - \eta_u)]} \quad (1.29)$$

where ϵ_P^* is the baseline plasma ion energy cost and is defined as

$$\epsilon_P^* = \frac{\epsilon_0 + \epsilon_M}{1 - [(V_C + \epsilon_M)/V_D]} \quad (1.30)$$

Here ϵ_0 is the plasma ion energy cost associating with ionization and excitation only, ϵ_M is the energy loss of secondary electrons, and V_C is the potential from which electrons are accelerated to become primaries. Also, C_0 describes the degree of primary electron and neutral interaction

$$C_0 = 4 \frac{\sigma_0 l_e}{e v_0 A_g \phi_0} \quad (1.31)$$

where l_e is the primary electron containment length, σ_0 is the total inelastic collision cross section for primary electron-neutral collisions, e is the elementary charge, v_0 is the neutral velocity, A_g is the grid area, and ϕ_0 is the grid transparency to neutral atoms. The variable ϕ_0 is the ratio of the entire number of grid holes areas over the entire area of the grid and is usually 70%. Using 1.29 and putting it back into 1.28 gets a more detailed version

$$\epsilon_B = \frac{\epsilon_P^*}{1 - \exp[-C_0 \dot{m} (1 - \eta_u)]} + \frac{f_C}{f_B} V_D \quad (1.32)$$

Equation 1.32 suggests that thruster performance is based on only four configuration parameters, ϵ_P^* , C_0 , f_B , and f_C and two operating parameters $\dot{m}$ and V_D . Also, this equation suggests that through the variable C_0 , the plasma ion cost depends on the following, taken from [2]

1. The propellant, which determines the total inelastic collision cross section σ_0 , and atomic mass, which affects the neutral velocity v_0
2. The wall temperature, which affects the neutral velocity.
3. The transparency of the grids to neutrals, ϕ_0
4. The area through which the beam is extracted, A_g
5. The discharge voltage, which determines the primary electron energy and thus affects the value of the cross section σ_0

This model has been verified to be accurate by experiments. Therefore, this thesis will use this model to calculate a performance model based on output results from numerical simulations. The simulations are aimed at providing certain key parameters for equation 1.32.

1.5 Thesis Topic and Approach

Before using a thruster, it is always wise to test the system. However, the classical method of building an actual size system for operational testing is very costly. Therefore, numerical analysis is a logical tool to approximate performance behavior prior to the design process. *The goal of this thesis is to develop a method to simulate ion thruster discharge chambers using 3D PIC MCC and commercially available software. Also, test cases with varying discharge chamber sizes will be used to both validate the simulation process and to develop a performance model based on energy cost to generate beam ions versus discharge chamber size.* This performance model will describe the changes in the energy cost of the thruster based on discharge chamber size changes.

The simulation process starts by meshing the discharge chambers using a program called Patran. This program converts the surface definitions into nodes and connecting these nodes into triangular surfaces to be recognized by the simulation program. Then, the magnetic field is generated using MaxwellSV. The two output data formats will then be converted to Tecplot data format since the simulation program requires that format for input files. The simulation program used in this thesis is the Coliseum project under development by Dr. Wang and his research students. This program has the full 3D PIC MCC capability. Even though it is MCC capable, this subroutine only does charge exchange collision. Electron impact ionization is the primary means of ionization in ion thrusters and is the only means of ionization in the simulation model. Therefore, the MCC subroutine is adapted to be ionization collision capable. Results from the simulation runs are visualized by Tecplot.

Chapter 2

Simulation model

The main question for this thesis project to answer is one of performance behavior due to discharge chamber size change. To do this, many ideas and concepts have to be accounted for and joined together. There is a presence of a static magnetic field inside the chamber so a method of field generation has to be found. In order to numerically see a structure, such as the discharge chamber, it has to be meshed into node positions. These nodes then have to connect into surfaces, like tetrahedral shapes, to form the desired structures. After the setup stage, the mesh and data are input into a 3D Particle-In-Cell (PIC) Monte Carlo Collision (MCC) code. The result of the simulation is then viewed with visualization software. Also, performance characteristics of the thruster can be extracted from the output data.

2.1 Particle-In-Cell Code

In a plasma, each charged particle affects, in one way or another, every other charged particle. In order to accurately model plasma particles movement, this effect has to be accounted for. The most basic method is to weight each particle against every other particle in the domain, but this is both time consuming and not necessary. A method was developed called Particle In Cell (PIC) method to treat a charged environment in term of basic properties such as number density, charge density, and field density. Information is calculated and stored on node points so particles being tracked can weight themselves to their neighboring nodes instead of the entire domain, as in figure 2.1. In the PIC method, the simulation domain is broken down into small cubes where the corners are node points. Particles are moved according to the field information, such as magnetic and electric field strength, on the nodes. After the the particles move, their properties, such as velocity and charge density, are used to calculate and update the information stored on the node points. An example of how PIC behaves is as follows. Given field information on the nodes where $\vec{B}$ is the magnetic field and $\vec{E}$ is the

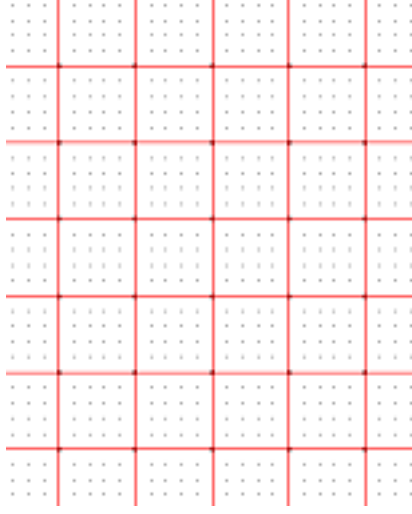


Figure 2.1: Checkered formation of PIC code

electric field. Particle movement following the Boris mover algorithm starts with the first half of the electric field acceleration

$$\vec{v}^- = \vec{v}^{n-\frac{1}{2}} + \frac{q\vec{E}}{m} \frac{\Delta t}{2} \quad (2.1)$$

The algorithm stops here in order to take care of the acceleration due to the magnetic field before completing the process. The magnetic acceleration is defined as

$$\vec{v}_1 = \vec{v}^- + \vec{v}^- \times \vec{t} \quad (2.2)$$

where

$$\vec{t} = -\hat{B} \tan \frac{\theta}{2} \quad (2.3)$$

Here θ is the rotation angle of the particle and is defined as

$$\theta = \Omega_c \Delta t \quad (2.4)$$

where Ω_c is the gyro frequency

$$\Omega_c = \frac{q\vec{B}}{m} \quad (2.5)$$

The other part of the magnetic acceleration is

$$\vec{v}^+ = \vec{v}^- + \vec{v}_1 \times \vec{s} \quad (2.6)$$

where

$$\vec{s} = -\hat{B} \sin \theta \quad (2.7)$$

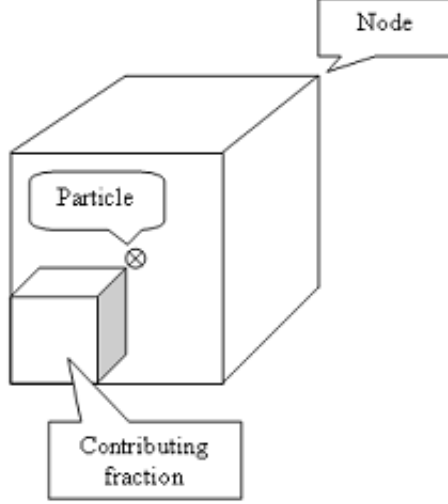


Figure 2.2: Force weighting where the contribution fraction is the fraction of the property of the node weighted onto the labeled node. This fraction is the contribution fraction over the entire cube space.

The second part of the electric acceleration to complete the process is

$$\vec{v}^{n+\frac{1}{2}} = \vec{v}^+ + \frac{q\vec{E}}{m} \frac{\Delta t}{2} \quad (2.8)$$

The placement is then changed according to

$$\vec{x}^{n+1} = \vec{x}^n + \vec{v}^{n+\frac{1}{2}} \Delta t \quad (2.9)$$

This method uses the leapfrog idea of half time stepping velocity and full time stepping displacement. Leapfrogging has more accuracy because displacement is approximated using velocities that are only half time step behind and in front therefore creating a smaller gap than using a full step each way. Now that particles have moved, the domain is no longer the same necessitating the need to update the nodal information. Number density and charge density are gathered through force weighting. This method weights particle properties to nodes by volumetric fractions, as shown in figure 2.2. After the densities are weighted onto the nodes, the electric field is calculated from the gradient of the charge density

$$\vec{E} = -\nabla \rho \quad (2.10)$$

If the magnetic field is not constant, then it would also need to be recalculated. Then the process repeats itself and that is how PIC goes about simulating a plasma.

2.2 Monte Carlo Collision

The Monte Carlo Collision (MCC) model used is based on Dr. Rodger J. Biasca's PhD dissertation [7]. This model is best described in reference [7], but this section will try to explain the concept as the author understands it and how it is used in the simulation. The MCC model is valid for the situation that the density of the neutrals is much larger than the density of the electrons population.

$$n_n \gg n_e \quad (2.11)$$

Hence, the simulation does not keep track of neutral particles.

The MCC calculation is divided into two phases. First the code determines whether there is actually a collision. Second it determines what to do with the energy and momentum of the particles in the collision and to produce new particles. The first starts by defining a collision frequency

$$\nu = \sigma v n_n \quad (2.12)$$

where σ is the collision cross section of the neutral particle, v is the relative velocity of the two colliding particles, and n_n is the neutral density. Then, a collision probability is calculated as

$$P = 1 - e^{-\nu \Delta t} \quad (2.13)$$

where Δt is the time step of the simulation. The term $\nu \Delta t$ has to satisfy

$$\nu \Delta t \ll 1 \quad (2.14)$$

for this method to work. If not, then the time step has to be reduced. To determine whether a collision between the particle and the cloud of density n has occurred, the collision probability is compared with a random number evenly distributed between 0 and 1. A collision has occurred if the random number is less than the collision probability.

$$p_{ran} < P \quad (2.15)$$

Since $\nu \Delta t$ is a lot less than 1, this means that the collision probability is very small ensuring the fact that collisional processes do not occur too often. The Monte Carlo aspect comes from the comparison of the collision probability with a random number. If there is no collision, go to the next particle and

repeat the process. If there is a collision, proceed to the second phase.

The second phase of MCC involves the process of how to collide the particles in a way that energy and momentum are conserved. The colliding particle is the electron that is running into a background of neutrals. The collision is calculated first by transforming the three dimensional space into a two dimensional collision space where the collision is assumed to be head on to ensure the particles will still be in the collisional plane afterward. The particles are then collided, scattered, then transformed back into three dimensional space. The normal unit vector to this two dimensional plane with respect to the two colliding particles is

$$\vec{n} = \frac{\vec{V}_{e1} \times \vec{V}_n}{\|\vec{V}_{e1} \times \vec{V}_n\|} \quad (2.16)$$

where the subscript $e1$ denotes that this is a first electron and n stands for neutral. Now, the velocities are transformed to the two dimensional collision plane as

$$\hat{V}_{e1} = \bar{U} \vec{V}_{e1} \quad (2.17)$$

$$\hat{V}_n = \bar{U} \vec{V}_n \quad (2.18)$$

where $\bar{U}$ is the transformation matrix from three dimensional space to the two dimensional collision plane. Also, define θ as the angle between $\vec{n}$ and the original z-axis, and ϕ as the angle between the projection of $\vec{n}$ onto the x-y plane and the x-axis. Define the transformation matrix as

$$\bar{U} = \begin{pmatrix} \cos \phi \cos \theta & -\sin \theta & \sin \theta \cos \phi \\ \sin \phi \cos \theta & \cos \theta & \sin \theta \sin \phi \\ -\sin \theta & 0 & \cos \theta \end{pmatrix}$$

Energy and momentum are conserved with respect to the center of mass frame of reference. The velocity of this frame is defined as

$$\vec{V}_c = \frac{m_n \hat{V} + m_e \hat{V}_{e1}}{m_n + m_e} \quad (2.19)$$

Conservation of momentum is defined as

$$m_e \hat{V}'_{e1} + m_e \hat{V}'_{e2} + m_i \hat{V}'_i = 0 \quad (2.20)$$

where i denotes ion. The conservation of energy equation is defined as

$$\frac{1}{2} m_n \hat{V}_n^2 + \frac{1}{2} m_e \hat{V}_{e1}^2 - (m_n + m_e) V_c^2 = \frac{1}{2} m_i \hat{V}_i'^2 + \frac{1}{2} m_e \hat{V}_{e1}'^2 + \frac{1}{2} m_e \hat{V}_{e2}'^2 + e \phi_{ion} \quad (2.21)$$

where $e2$ denotes a second electron that was produced from the collision and prime denotes the component after the collision. Therefore, two particles

come into the collision and three particles with different velocities come out. In order to give the three particles velocities, the two conservation equations have to be solved. To do this, three random angles (α, β, γ) are generated to represent the scattering angle of each of the particles in the collision plane. The velocities of particles after a collision are

$$\hat{V}'_{e1} = \frac{m_i \sin(\beta - \gamma)}{m_e \sin(\alpha - \beta)} \hat{V}'_i = A \hat{V}'_i \quad (2.22)$$

$$\hat{V}'_{e2} = -\frac{m_i \sin(\alpha - \gamma)}{m_e \sin(\alpha - \beta)} \hat{V}'_i = B \hat{V}'_i \quad (2.23)$$

$$\hat{V}'_i = \left(\frac{\hat{V}'_{e1} + \frac{m_n}{m_e} \hat{V}'_{e2} - \left(1 + \frac{m_n}{m_e}\right) V_c^2 - \frac{2e\phi_{ion}}{m_e}}{A^2 + B^2 + \frac{m_i}{m_e}} \right)^{\frac{1}{2}} \quad (2.24)$$

generalizing these three equations to velocities in the collision plane, center of mass frame are

$$\hat{V}'_j = \left(\hat{V}'_j \cos \theta_j, \hat{V}'_j \sin \theta_j \right) \quad (2.25)$$

where $j = e1, e2, i$ and $\theta = \alpha, \beta, \gamma$. Now, the velocities are converted back into the three dimensional space, or laboratory frame, in order for them to be used. This is done from the center of mass frame by

$$\hat{V}'_{j,L} = \hat{V}'_j + \vec{V}_c \quad (2.26)$$

as the result, we would get the following back from the MCC function

$$V'_{e1,L} = \bar{U}^T \hat{V}'_{e1,L} \quad (2.27)$$

$$V'_{e2,L} = \bar{U}^T \hat{V}'_{e2,L} \quad (2.28)$$

$$V'_{n,L} = \bar{U}^T \hat{V}'_{n,L} \quad (2.29)$$

where $\bar{U}^T$ is the transpose of the transformation matrix. So, in summary, the algorithm for MCC process is as follows

1. Get velocities for electron and neutral.
2. Calculate a collisional frequency and collision probability.
3. Generate a random number between 0 and 1
4. If random number is less than collision probability then there is a collision. Proceed further
5. Transfer the particles to a two dimensional collision plane
6. Proceed to collide the particles and assign new velocities accordingly
7. Convert the velocities back to three dimensional space
8. Return velocities

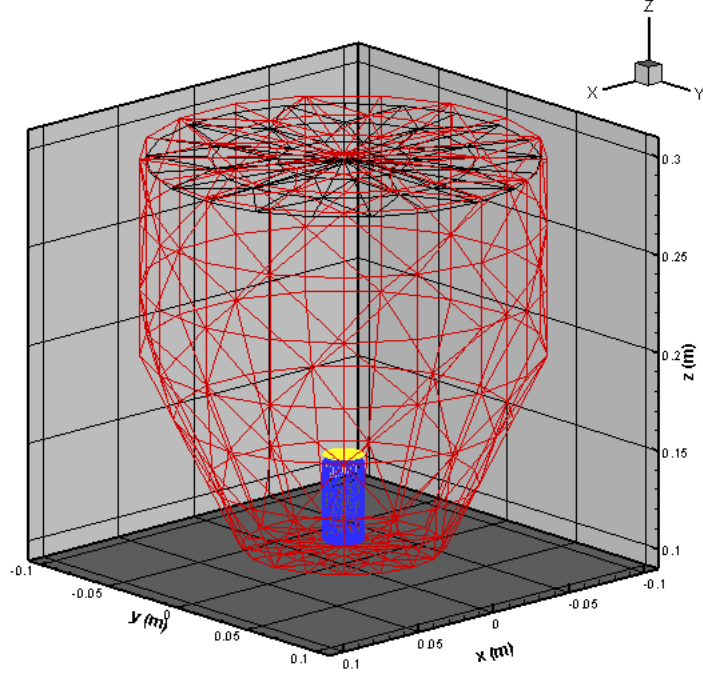


Figure 2.3: 3D meshed discharge chamber where the chamber cathode is in blue and the grid sink is in black. The source is placed one cell in front of the cathode and the sink is placed one cell before the grids. The cathode is placed one cell in front of the back chamber wall.

2.3 Mesh Generation

Structures need to be broken down and redefined in terms of meshes in order to be used in simulations. The program chosen for this task is PATRAN. PATRAN has a lot more capabilities than just meshing objects, but this is the only option that is used here. This section goes through the process of creating the discharge chamber structure and meshing it. Since this thesis only focuses on the discharge chamber, outside characteristics of an ion thruster need not be modeled. Therefore, only the basic shape of the chamber, namely the cone and cylinder shape, was modeled. Inside, the cathode assembly is modeled as a cylinder with a gap to the rear chamber surface. The meshed shape is shown in figure 2.3. Here, the source surface is simulating the exit plane of a cathode where electrons are injected into the chamber. The grid surface acts as a sink to either remove the particle going through the grids or remove particle charges hitting the grids.

The cathode is placed a small distance away from the back chamber wall due to the mesh recognition conflict in Coliseum. Every surface element is

given a surface vector and this vector determines whether space is a solid or empty space. If the the space is in the direction of the surface vector, it is empty space. If the space is opposite of the vector, it is a solid. Therefore, if the cathode surface flushes with the back chamber wall surface, the program gets confused about space definition. Similar explanation goes for the placement of the cathode source to justify its placement in front of the cathode surface. The nodes at the cathode surface can not also be defined as source nodes. The sink surface is placed behind back of the chamber. This sink simulates the particles lost to the grids. When particles come in contact with this sink, they are taken out of the simulation. The reasoning is that if electrons hit the grids, they will get absorbed. In the case of ions, they have a 70% chance of being used as thrust when they get to the grids and 30% of hitting the grid and reflecting back as neutrals. Either way, these ions are lost to the simulation.

2.4 Magnetic field generation

Methods of conversion between raw mesh data from PATRAN to TECPLOT format and 2D magnetic field data to 3D TECPLOT format are described in APPENDIX A and APPENDIX B. The first attempt of configuring a shape for the simulation thrusters is shown in figure 2.4. This configuration is built on simplicity as the cone and the straight section of the thruster are each half the length of the grid. Also, the discharge cathode assembly length is 20% of the grid length and the gaps being 1.7% of the grid length. As shown in the meshed figure, the cathode exit plane (the right side of the cathode cylinder) is well behind the middle magnet ring. This creates a problem for the electrons exiting the cathode. What they see when get injected into the chamber is the presence of magnetic field lines generated by the middle magnet ring and the ring behind the chamber. Most of the electrons will get trapped by these field lines and therefore never traverse toward the main section of the chamber, shown in figure 2.5.

The reason why the electron distribution is in a cone shape is because they could not truly go back to the back magnet since the high electron density aft of the cathode repels them toward the mid magnet. Because of this, the electrons are trapped in this region. Since it is desired to have electron distribution in the latter part of the chamber, the design has to be changed. The most obvious solution is to inject electrons in a way that they will see the magnetic field lines generated by the middle and the front part of the thruster. To do this, the cone section length is reduced by 50% making it only 25% of the grid length. With the cathode length being 20% of the grid length, injected electrons will definately see the field lines from the middle and the front magnets. The final shape design is shown in figure 2.7.

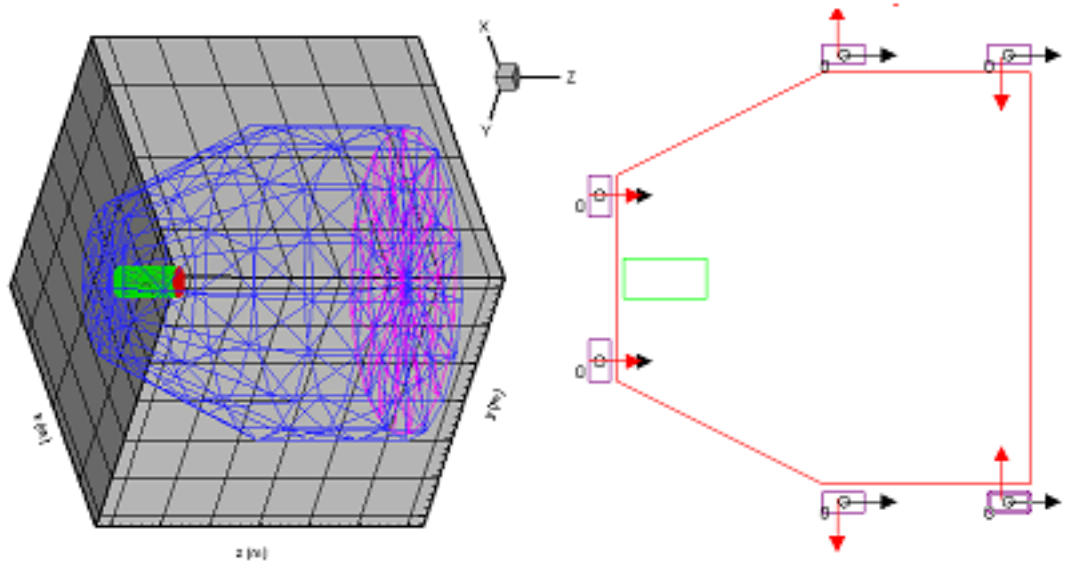


Figure 2.4: Original shape and magnet configuration for simulation thruster. On the left is the meshed surface for the chamber with blue mesh representing the chamber wall. The green cylinder is the discharge cathode assembly with the red surface as the source surface where electrons will be injected into the chamber. Screen grid is the purple surface where particles will be taken out of the simulation on contact. The figure on the right is the 2D magnet configuration with the green cylinder also representing the discharge cathode assembly. The red arrows represent magnetic field directions.

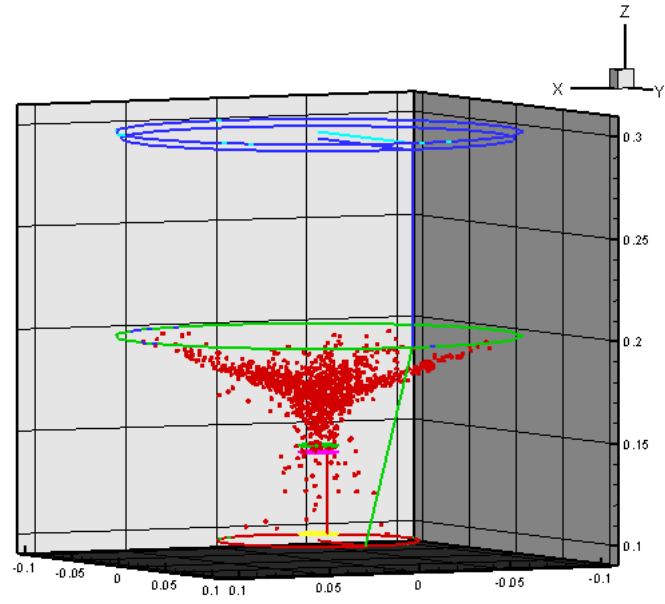


Figure 2.5: Electrons are trapped by the magnetic field lines generated between the middle and back magnet rings.

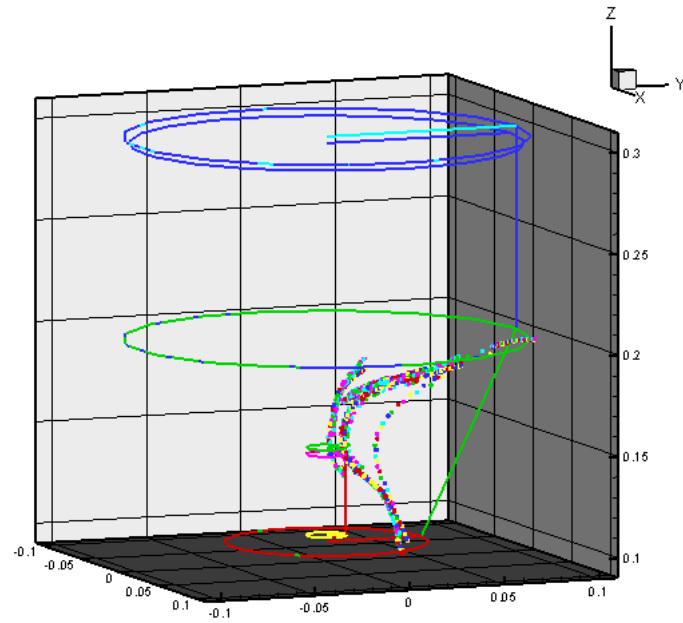


Figure 2.6: This is a trajectory plot of two electrons traveling back and forth between the back and middle magnet rings.

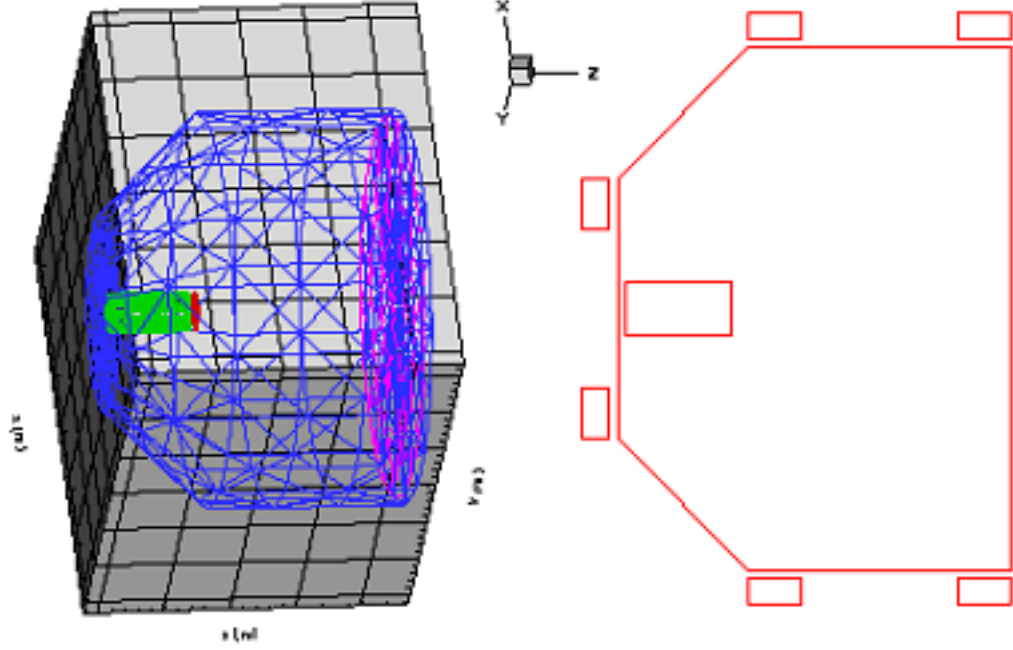


Figure 2.7: Cone section length reduced by half to partially flush the source surface with the latter section of the chamber.

Now that the shape is finalized, attention is focused on the magnetic field. Originally, the field configuration is generated by MaxwellSV and adapted to values according to [12]. Data from this paper suggests that the magnetic field strength around the middle of the chamber is around 50 Gauss, or 5×10^5 Tesla. The value of magnetic field strength directly from MaxwellSV around the middle of the chamber is $\approx 0.01T$. This value needs to be converted to 5×10^5T , to adapt it to the magnitude in reference [12], therefore a conversion factor of 5×10^7 is used. Since the relative field strength from point to point in the chamber is correctly modeled by MaxwellSV, the conversion factor is applied to all data values. After a simulation run, the resulting electron distribution is shown in figure 2.8 The magnetic field shown appears to be too strong since the electrons are focused into a narrow beam toward the grid. Electrons must be allow to move toward the cusps. This is to allow a fraction of electrons to be trapped in the chamber. To do this, the magnetic field strength needs to be relaxed. It has been found that the magnetic strength directly from the solid magnets, in its unaltered state, does not need to be converted. Therefore, the conversion factor stated above is taken out of the process. The final magnet configuration is shown in figure 2.9 with magnet size reduced to further relax the magnetic field strength. The magnetic field distribution contour plot generated from this

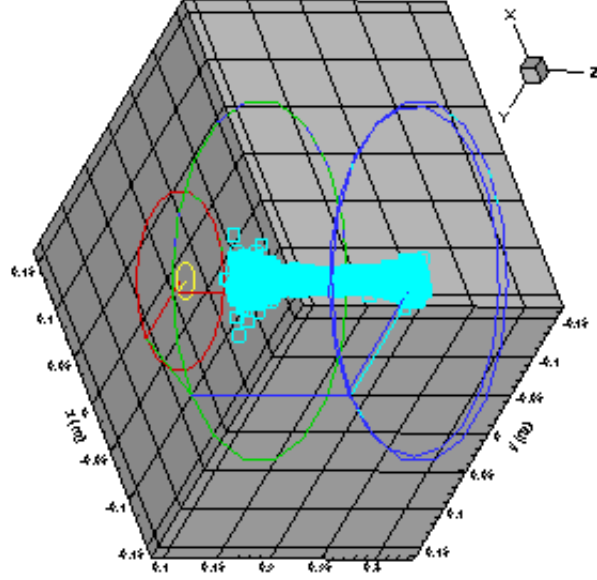


Figure 2.8: Electron distribution using magnetic field conversion factor of 1×10^7 . Here, the magnetic field is so strong that electrons are focused into a beam toward the grids.

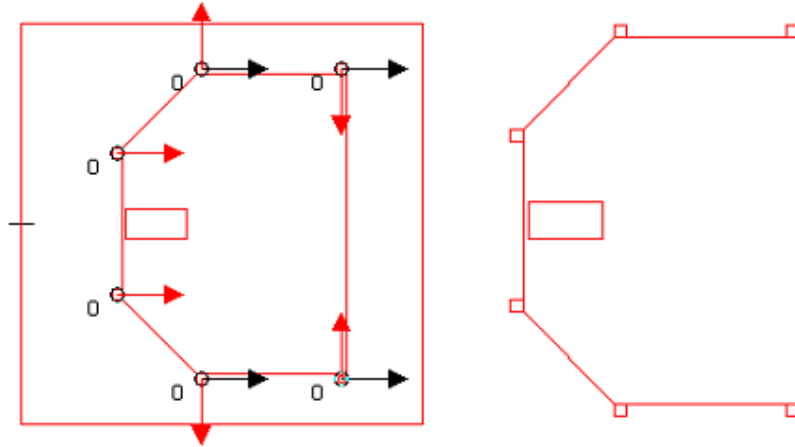


Figure 2.9: Final magnets configuration with magnets size reduced.

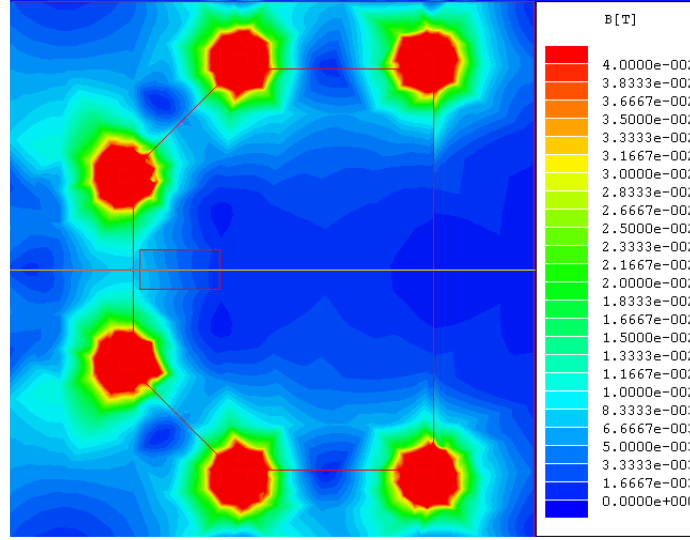


Figure 2.10: Magnetic field contour plot generated from the final magnet configuration. The deep blue region in the middle is the low field region. All values are in Tesla.

configuration is shown in figure 2.10, The shape of the low field region in the middle of the chamber is very similar to the distribution plot of [12], verifying the field validity.

Chapter 3

Discharge chamber simulation

3.1 Model Setup

The simulations done in this thesis follows a model with information gathered from several sources. Since no single paper can be found that has complete detailed information on all the aspects of thruster operations and parameters, more than one source is necessary. The following aspects of thruster operation are needed to build a model for the simulations.

- Grid area, A_g
- Magnetic field configuration
- Discharge Cathode Assembly exit area, A_D
- Discharge Cathode Assembly current, I_D
- Discharge Cathode Assembly voltage, V_D
- Discharge Cathode Assembly electron energy, E_D

Useful data on operational parameters, such as I_D and V_D , are taken from an experiment done by Herman and Gallimore [9]. Since no mention is made of any particular discharge cathode assembly used in the experiment, a 3.2mm cathode assembly, referenced from similar experiment or simulation of [10] and [11], was used. From I_D and V_D , the electron density at the exit area of the cathode is calculated from the charge density (J) equation

$$J = env = I_D/A_D \quad (3.1)$$

where A_D was derived from the cathode exit radius that is half of the cathode outer diameter of 3.2mm, $r_D = 1.6[mm]$, making

$$A_D = \pi r_D^2 = 8.04 \times 10^{-6}[m^2] \quad (3.2)$$

This leads to a number density of electrons aft of the cathode exit plane as

$$n_e = 1.57 \times 10^{18} [m^{-3}] \quad (3.3)$$

which is similar to data from [11]. From this number density, the plasma frequency and the Debye length can be found

$$\omega_p = \left(\frac{n_e e^2}{\epsilon_0 m} \right)^{\frac{1}{2}} = 7.06 \times 10^{10} [Hz] \quad (3.4)$$

where $\epsilon_0 = 8.85 \times 10^{-12}$, $e = 1.6 \times 10^{-19} C$ is the electron charge, and $m = 0.91 \times 10^{-30} kg$ is the electron mass.

$$\lambda_D = 740 \left(\frac{T_{eV}}{n_e} \right)^{\frac{1}{2}} = 1.25 \times 10^{-5} m \quad (3.5)$$

where $T_{eV} = 4.5$ [11] is defined in electron volts and n_e is defined in $[cm^{-3}]$. Careful considerations for these units were taken cared of and final result is converted to standard unit of meter. From this value, the ratio of grid length over debye length is

$$\frac{d_g}{\lambda_D} = \frac{0.3[m]}{1.25 \times 10^{-5}[m]} = 24000 \quad (3.6)$$

Velocity of the exit electrons can be calculated from the energy conservation formula

$$\frac{1}{2} m v^2 = e V_D \quad (3.7)$$

which leads to

$$v = 2.995 \times 10^6 [m/s] \quad (3.8)$$

Another important variable which can now be calculated is the electron ionization mean free path, λ_{mfp} . This variable is dependent on the neutral density. Since the calculated number density of electron matches values in [11], number density for neutral is gathered from that same source to be

$$n_n = 4 \times 10^{20} [m^{-3}] \quad (3.9)$$

Now λ_{mfp} can be calculated as

$$\lambda_{mfp} = \frac{1}{n_n \sigma} = 4.5 \times 10^{-2} [m] \quad (3.10)$$

where σ is taken as the maximum ionization cross section between xenon and electron, $\sigma = 5.5 \times 10^{-20} [m^2]$. A relationship between the grid diameter and ionization mean free path can now be derived

$$\frac{d_g}{\lambda_{mfp}} = \frac{0.3[m]}{4.5 \times 10^{-2}[m]} \approx 6.7 \quad (3.11)$$

This means that the electrons will only travel around a sixth of the grid length before it hits and ionizes a neutral assuring that ionization will definitely occur. Therefore, this ratio will be kept constant for the simulation cases also. Another electron characteristic, gyroradius, needs to be kept track of. This is

$$\rho_l = \frac{mv_{\perp}}{eB} = 1.7 \times 10^{-3}[m] \quad (3.12)$$

where the magnetic field strength, $B = 0.01T$, is chosen to be around the middle of the chamber of a permanent magnetic configuration. The value for the gyroradius above represents the largest possible considering maximum input velocity used as velocity perpendicular to the magnetic field. In reality, only part of this electron velocity is perpendicular to the field making the gyroradius even smaller. The basic parameters for the model case are now known.

The last thing to take care of is the method of ion generation. This requires electron colliding with neutrals ionizing them. This aspect of the thesis is the most time consuming, taking up most of a whole semester to complete. At the time this thesis started, Coliseum did not have all the options that are required to do what this thesis is set out to do. Coliseum had a version of MCC built in, but it could only take care of charge exchange case. A MCC version that has the ability to do ionization has been translated into C++, but it is not compatible with the current Coliseum version and therefore commented out of the project. In order to reincorporate this version of MCC back into Coliseum, the data structure and methodology of the project has to be understood and used. This and Coliseum changing on a regular basis were the main factors of why reincorporating MCC back into Coliseum was so time consuming. But in the end, MCC does what it was intended to do.

3.2 Scaling

In order to correctly model electron motions, the length scale of the simulation has to be comparable to that of the Debye length, otherwise the resolution is not sufficient to resolve the forces applied to electrons correctly. Therefore, the most logical method of plasma simulation is to have a mesh with each cell length equivalent to the Debye length. Unfortunately, due to the limitation of computational power, only meshes of around thirty nodes per dimension can reasonably be used. There are two ways of approaching this problem. One is to reduce the cell length down to the original Debye length of $1.25 \times 10^{-5}m$ and have the grid length as 30 Debye lengths. The other approach is to decrease the number density of electron down to stretch the Debye length up to 1 cm. Both approaches could provide the same result if done correctly. The stretching approach is adapted for this thesis

simulations. Parameters for the simulations are calculated as follows. Since the new Debye length is being stretched to 1 cm, back solving the Debye length equation gets the number density for electron

$$\lambda_D = \left(\frac{TeV}{n_e}\right)^{\frac{1}{2}} = 1[cm] \quad (3.13)$$

$$n_e = 2.46 \times 10^{12}[m^{-3}] \quad (3.14)$$

This is a significant decrease in the number density, but it is necessary to increase the debye length by an order of magnitude of 3. The number density of neutrals above is held constant throughout the chamber. This means that the number density of the neutrals is not simulated, rather it is just a constant to be used in the calculations. The reason why this is done is due to the processing power limitation. If the neutral particles are kept tracked of, this presents a number larger than the current computer memory ability.

Now that the debye length is 1 cm, the original grid length of 30 cm is conserved and the domain is divided into cells with 1 cm length in each dimension. The cathode size is set as 3 cell lengths by 6 cell lengths to match with figure 1.2. This is not to scale as the 3.2 mm discharge cathode assembly is a lot smaller. Since the only thing that really matters is the electron mass flow rate out of the cathode, size difference does not make a significant issue. Discharge cathode assembly electron mass flow rate is calculated from back solving the charge density equation, keeping the number density in front of the cathode the same. From the charge density equation, the new current is found and converted to mass flow rate in term of $\frac{kg}{s}$

$$J = en_e v = I_D / A_D \quad (3.15)$$

where $A_D = 8.04 \times 10^{-6} m^2$ and v is given from the beginning of this chapter as $2.995 \times 10^6 m/s$, getting

$$I_D = 9.49 \times 10^{-6}[A] \quad (3.16)$$

converting to kg/s

$$\dot{m}_D = I_D[A] * 6.24 \times 10^{18} [\frac{C/s}{A}] * 0.91 \times 10^{-30} [\frac{kg}{C}] = 5.39 \times 10^{-17} [kg/s] \quad (3.17)$$

Also, the time step has to be small enough to be able to keep the electron inside the debye length after each time step. This is to make sure that electrons do not run outside of resolved regions.

$$dt = \frac{\lambda_D}{v} \approx 3 \times 10^{-9}[s] \quad (3.18)$$

The purpose of this thesis is to study the effect of chamber size on energy cost of thruster to create beam ions, therefore the parameters above will be kept constant as only the chamber size will change. In summary, the operational parameters are shown in table 3.1.

Debye length	1[cm]
n_n	$2 \times 10^{20}[m^{-3}]$
n_e	$2.46 \times 10^{12}[m^{-3}]$
v	$2.995 \times 10^6[m/s]$
Te	4.5[eV]
$\dot{m}_D$	$5.39 \times 10^{-17}[kg/s]$
dt	$3 \times 10^{-9}[s]$

Table 3.1: Input parameters for simulation cases

3.3 Results

The results presented in this thesis are only preliminary due to the reduced densities used due to the computing limitation. The same process stated above could be used to fully simulate ion thruster discharge chambers if more computing power is available to fully incorporate the much higher densities.

3.3.1 Analysis and validation of simulation results (30 cm case)

Preliminary results have suggested that the simulation setup is correct. Further analysis of data output from the simulations will now be analyzed in more detail in this chapter. The main focus will still be on the 30 cm case to validate the data from simulation with various sources. Afterward, results from other simulation runs will be shown and then a performance curve is then derived from these results. A good starting point is to look at the chamber before any particle is injected into it. A plot of the electric potential is shown in figure 3.1. As stated before, the chamber wall is set at 1100 V, the cathode is set at 1075 V, and the grid is set at 1070 V. The cathode is the blue vertical cylinder and the grid is the blue horizontal line toward the top of the plot. Due to this difference in object charges, there exists field gradients in the chamber, all of which points toward the middle of the chamber. Imagine an electron that is injected into the chamber and consider only the electric field. The first thing that it sees is the presence of an electric field gradient up-sloping toward the middle of the chamber, electrons moves up in electric field gradient due to their negative charge. Therefore, they will speed up according to the energy conservation equation

$$E = \frac{1}{2}mv^2 = eV \quad (3.19)$$

where m is the electron mass, e is the electron charge, V is the difference in the voltage, and v is the electron velocity. This electron also sees the presence of an electric field up-sloping toward the sides, but as it goes toward the sides, it will get trapped by the magnetic field. Therefore this

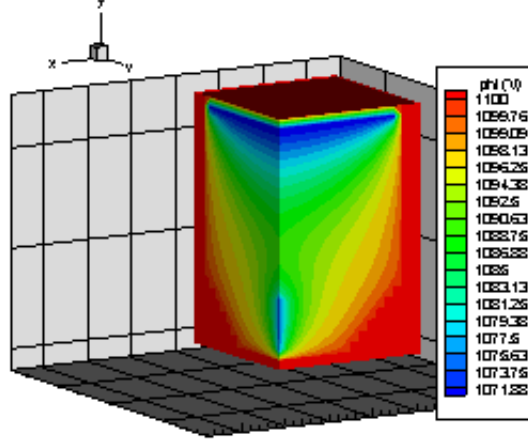


Figure 3.1: Electric potential distribution before simulation starts. There is a gradient toward the middle of the chamber. This gradient acts to help trap the electrons in the low magnetic field region also in the middle of the chamber. All values in volts.

example will not concern with radial motion, but only axial motion. But as the electron moves toward the grid, it finds that somewhere in the middle of the chamber, the gradient turns against its progress. This down-sloping gradient effectively mirrors it backward toward the middle of the chamber again. This is why it is important to set the potentials in this manner.

Aside from the low magnetic field region in the middle of the thruster, electron distribution will depend on the magnetic field. A plot of magnetic field lines distribution is shown in figure 3.2. In this figure, the magnetic field lines at the cylindrical section of the chamber has a "C" shape suggesting that electrons will be concentrated into an "I" beam formation. At the cone section of the chamber, field lines connect the magnet ring behind the chamber with that at the middle magnet ring, also suggesting the electron distribution in this manner. In the region close to the cathode exit plane, it is interesting to see that the field distribution expands outward getting a bit fatter. This is because at the X-Y plane of the middle magnet ring, field strength is greater and therefore the field lines are trying to get to the ring. This can also be expected of the electron distribution at this region. Ions, with their mass greatly heavier than electrons, will not follow these fields since their gyroradius are larger than the chamber size, and so therefore cannot be predicted from looking at magnetic field lines distribution.

In order to prove that the electron motions are correctly resolved, by having a small enough simulation cell size, electron gyro radii must be seen

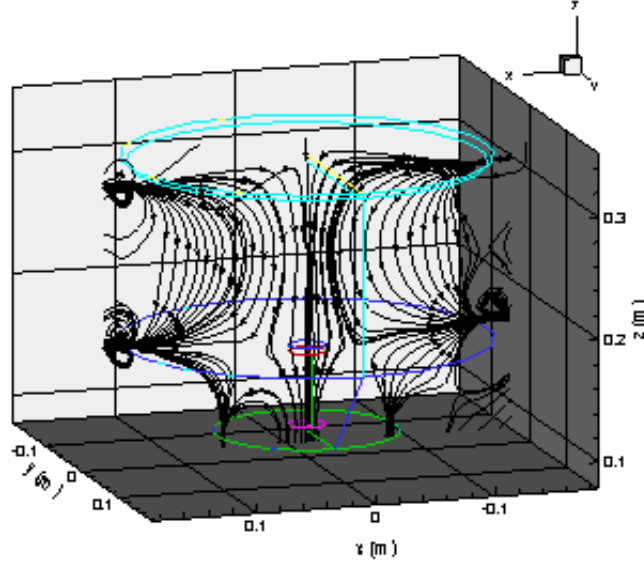


Figure 3.2: Magnetic field lines distribution within the discharge chamber. This distribution will determine the location of the electrons due to electron gyration motions and magnetic trapping effect.

as they are being simulated. The following result is from a 2000 time step run on a 30 cm diameter thruster with a cell length of 1 cm. Figure 3.3 shows a sample motion of two electrons. In the left case, time step from 1 to 50, it shows one electron gyrating along the middle magnetic field line and going to the grid. The second electron has lower kinetic energy toward the grid and therefore gets turned back by the down-sloping of the electric field. The right plot shows a case that happens between time step 1900 to 2000. Here, two electrons gyrate along a magnetic field line going toward the middle magnet ring. Since this electron has more kinetic energy than the change in field line density can stop, it is lost to the cusp produced by the middle magnet. From this result, *it has been shown that electron motion has been resolved.*

After a simulation run of 1000 time steps for a 30 cm thruster, the resulting particles population is shown in figure 3.4. As shown, the population for ion starts to converge around 250 time steps. The electron population still increases at the end of the simulation run, but does not seem to affect the population of the ion. An explanation for this continuing increase in the number of electrons can be from electrons generated from collisions. Every time there is a collision, an extra electron is created along side with the ion. Due to the fact that particles can co-exists within the same space, meaning

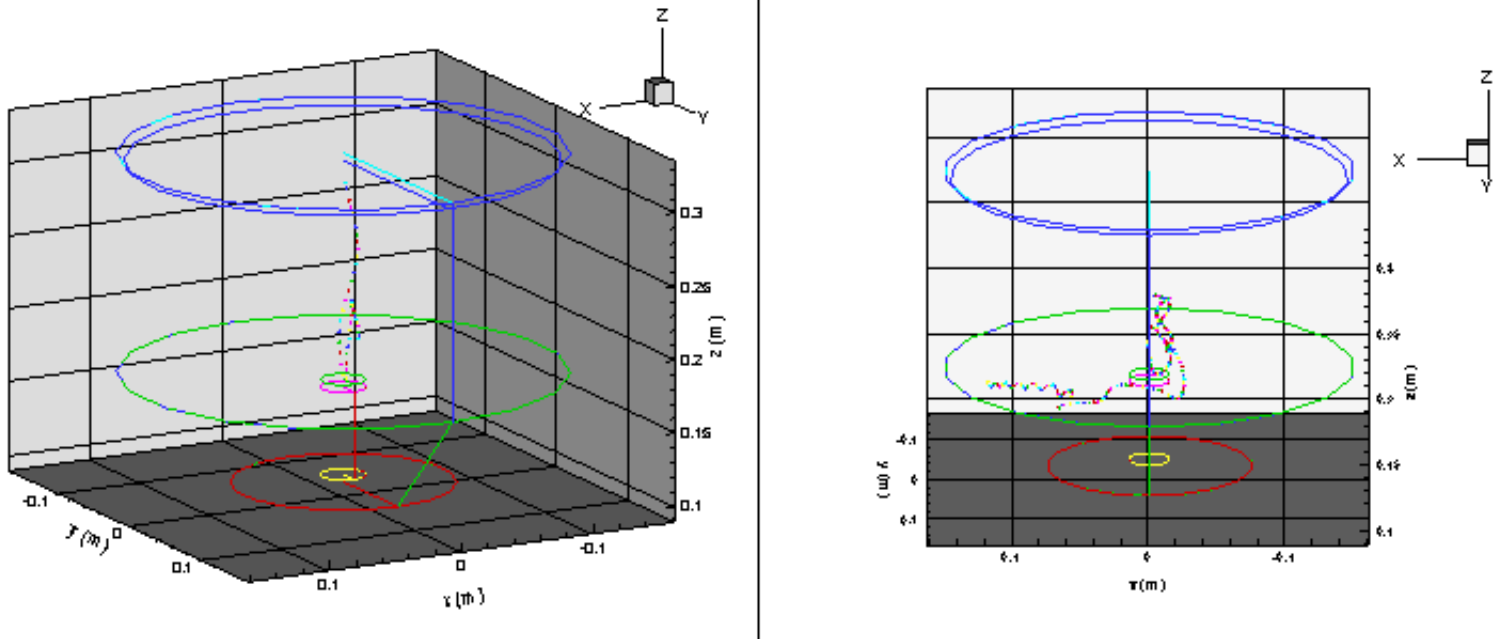


Figure 3.3: These two figures show a sample motion of two electrons, one going toward the grid (left) and one going toward the middle magnet ring (right). The gyro radii can be clearly seen.

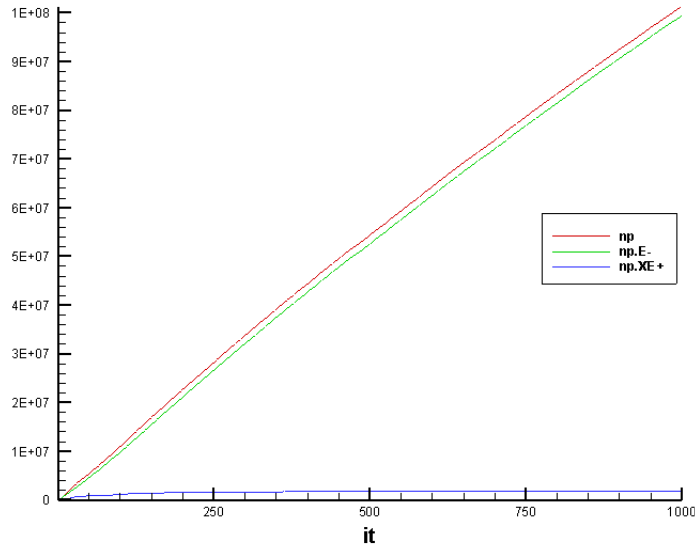


Figure 3.4: Particles population of 30 cm thruster after 1000 time steps. Ion population begins to converge after around 250 time steps. Even though electron population continues to rise, this does not seem to affect the ion population.

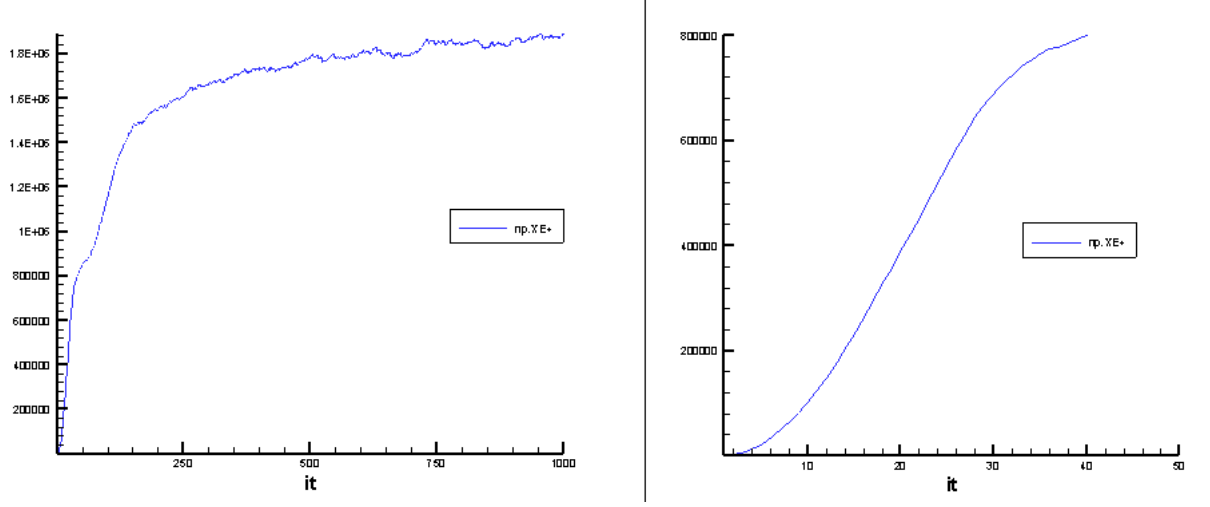


Figure 3.5: Ion population plot with only 50 time steps shown on the right figure. The first slope represents ionization before ions are lost to the surfaces.

that the simulation program does not consider inter-particle collisions other than for ionization, the newly created electrons can similarly be trapped by the fields. When ionization occurs, the program puts the three particles, initial electron, newly created ion, and newly created electron, at the same location just with different velocities. Since ions are the main focus of this simulation, because they create the thrust for the thruster, a closer look at the ion population is shown in figure 3.5. This figure shows definite convergence at 1000 time steps. A case run from 1000 to 2000 time steps has also shown that the number of ions in the domain does not change, but the number of electrons continue to increase in the same manner as in figure 3.4. Therefore, other cases will only run to 1000 time steps. About ionization rate, it would be interesting to know how the rate on paper compares to the actual rate in the simulation. On paper, the ionization rate is estimated, based on MCC calculation, first from the collisional frequency

$$\nu = \sigma v n = (5.5e - 20)(2.995e6)(1.9e20) = 31215387.5 \quad (3.20)$$

This value is calculated from initial input values and is in terms of collisions per second. The next step is to get the collision probability

$$P = 1 - \exp(-\nu dt) = 1 - \exp(-(31215387.5)(1e - 9)) = 0.031 \quad (3.21)$$

This means that every particle with the above parameters would only have a 3 percent chance of ionization collision. At time step around 35, the electron population, read from figure 3.5, is around $3e7$. This information coupled

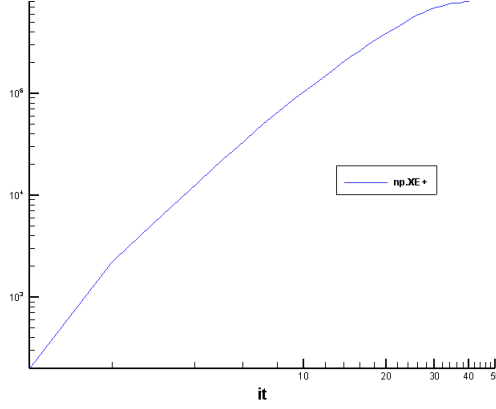


Figure 3.6: Ion population of a 30 cm thruster for the first 50 time steps

with the collision probability of around 3 percent and the time step size of $1e - 9$ seconds produces an ionization rate of

$$\nu \approx 9e14 \quad (3.22)$$

This number is in term of ions per second. Since the σ used in the previous calculation is the maximum value possible, meaning an energy of 100 eV, most of the electron population does not qualify. Therefore, one would expect the actual ionization rate from the simulation to be lower.

Now, to get the ionization rate from the simulation, the ion population plot for the first 50 time steps will be plotted in log-log scale, shown in figure 3.6. Two points are chosen to calculate the slope of the straight part of the plot, (2,332) and (16,260980).

$$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{260980 - 332}{16 - 2} \approx 18617 \quad (3.23)$$

This slope represents the number of ions created per time step, where one time step is $\approx 1e - 9$ seconds, a time history of the time step size is shown in figure 3.7. From the calculated ionization rate above, this results in an ionization rate from the simulation of

$$\nu_{simulation} \approx 1.8e13[ions/sec] \quad (3.24)$$

The ionization rates from calculation and simulation have a little less than two order of magnitude difference, which is acceptable considering not all electrons will have the kinetic energies of 100 eV, making their collisional cross section smaller than that used in the initial calculation.

For the same case as above, the electron distribution is shown in figure 3.8 and 3.9. Similar to calculated numbers prior to the simulation run,

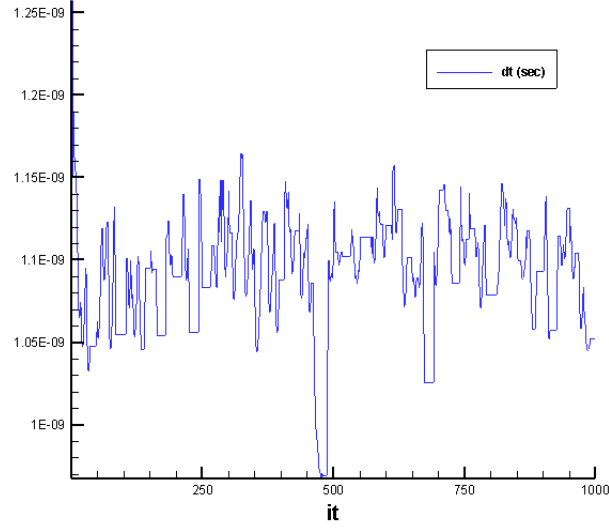


Figure 3.7: Time step size history of the 30 cm thruster simulation case. The size is around $1.1\text{e-}9$ seconds throughout the simulation.

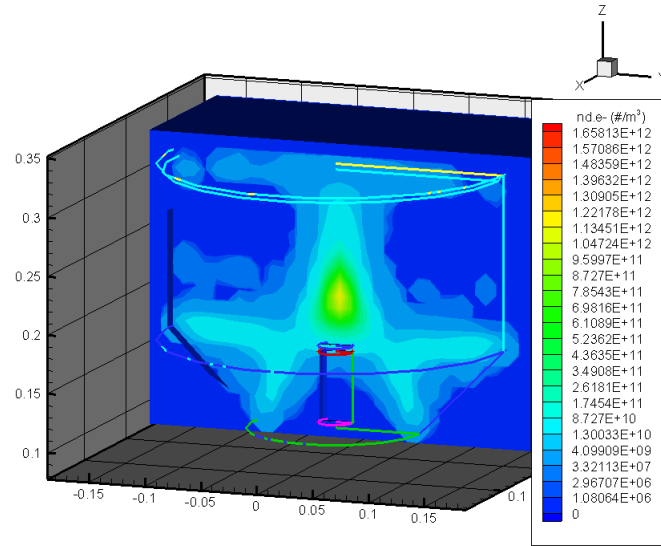


Figure 3.8: Electron density contour plot for the 30 cm thruster case after 1000 time steps. It is densest at the middle of the chamber where field strengths are at their lowest. The surrounding electron population outlines the magnetic field lines configuration.

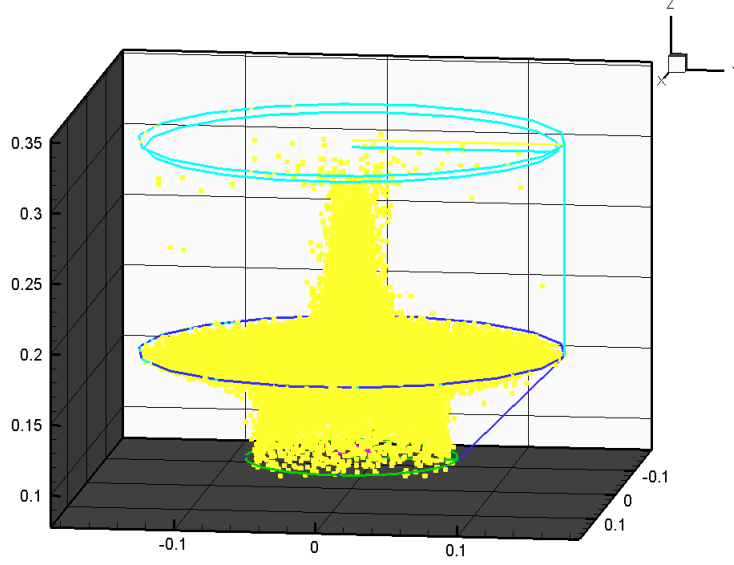


Figure 3.9: Electron distribution for the 30 cm thruster case after 1000 time steps. A 3D look at the population.

the densest part is aft of the cathode exit plane. This calculated density is $2.46e12 \text{ m}^{-3}$. From the contour plot, the density aft of the cathode exit plane ranges from $1e12$ to $1.6e12 \text{ m}^{-3}$. The simulation resulting density is very close to the calculated value. Even though there is a difference, it is not enough to invalidate the simulation result. Going away from the center axis of the chamber, electron motion appear to be governed more by the magnetic field line distribution. Upon comparison, one can see that the electron distribution matches with the magnetic field configuration in figure 3.2. This is because electrons gyrate along field lines. There is a lack of electrons in the regions near the chamber wall away from the magnet rings. This is due to the magnetic mirror effect. As electrons try to move toward the wall away from the cusps, they encounter a change in magnetic field density that results in a opposing force. This force prevents them from moving further on, therefore trapping them toward the center of the chamber. It is widely known that the more magnetic cusps that a chamber has, the more electrons will be lost to them.

The ion distribution for the 30 cm thruster case is shown in figure 3.10. The ion density plot shows that it is densest at the middle of the chamber and decreases radially outward. This behavior is expected since the electron density is highest in the middle of the chamber also. More electrons means higher ionization probability. This result is very similar to the result from Wirz and Katz [13]. In their result, the maximum density, which is also in

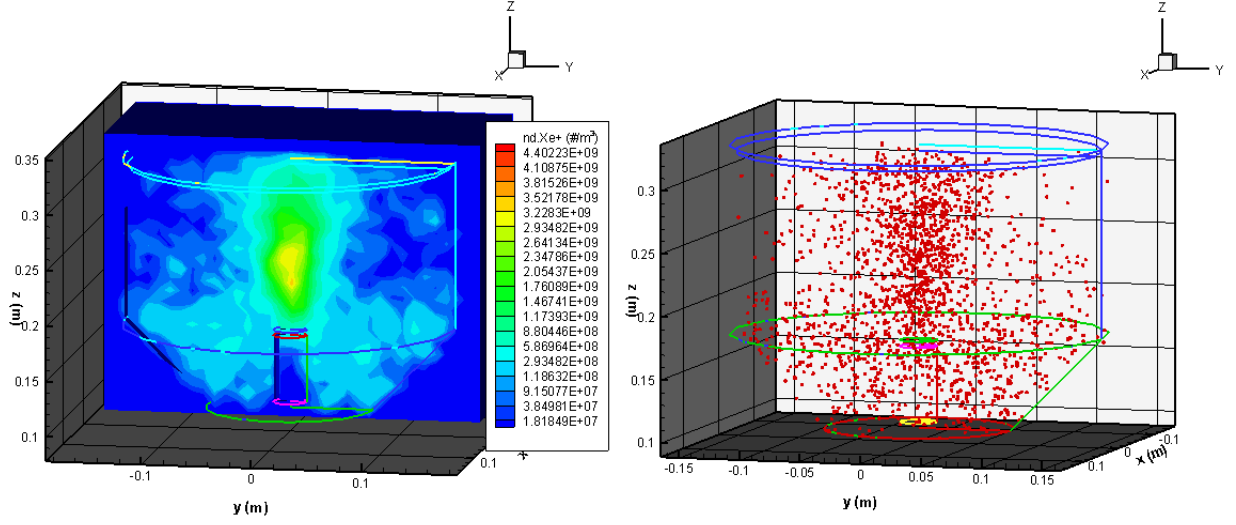


Figure 3.10: Ion density and distribution plot for 30 cm thruster case. Density decrease in the radial direction since electron density is highest in the middle.

the middle of the chamber, is on the order of 10^{18} while the result from this thesis is on the order of 10^9 . The reason for the smaller number from this thesis is the same as stated above about electron number density reduction. The case of ion density being at maximum along the axial direction of the thruster is a desired one. Since the density is still very high all the way to the grid suggests that more of them will get to the grid and therefore be used in thrust creation. Performance of an ion thruster can be characterized as the ratio of ions used for thrust versus ions lost to chamber wall. Therefore, the more that can be focused in the axial direction toward the grid, the better. This concludes the analysis and validation of simulation part of this case. Next, output data from other simulation runs will be introduced.

3.3.2 20 cm case

When the chamber size deviates from the 30 cm case, all the operating parameters remain the same. To make things simpler, the size of the magnet rings are also held constant. This causes a slight change in the magnetic field configuration, mainly the strength, this is shown in figure 3.11. The magnetic field strength for the 20 cm case is stronger than that of the 30 cm case. This is explained by the fact that the chamber size is smaller while the magnet size remains the same. From both the chamber size being smaller and the magnetic field being stronger, one would expect to find the electron distribution to be tighter more focused toward the middle. The resulting electron density plots, from a 1000 time steps simulation run, are shown in

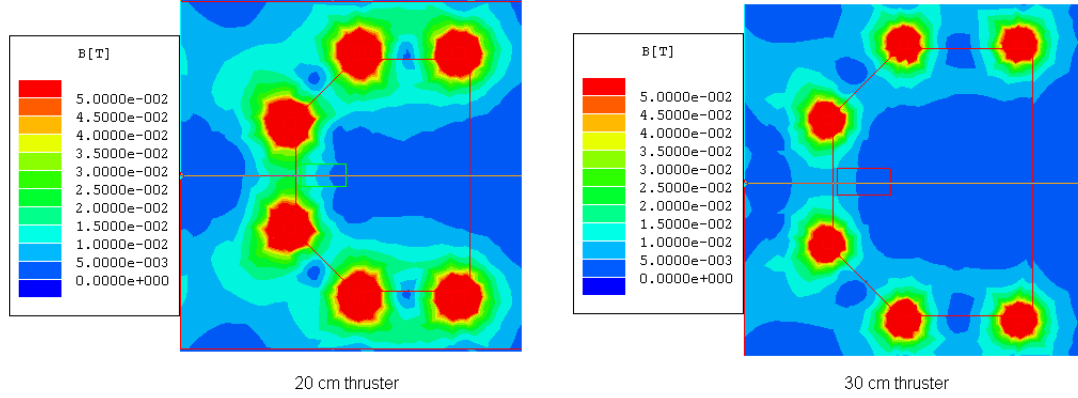


Figure 3.11: A comparison between a 20 cm and a 30 cm thruster magnetic field strength plot. The 20 cm thruster has a stronger overall field strength due to its smaller dimensions with the same magnet sizes.

figure 3.12. At the middle of the chamber aft of the cathode exit plane, the same density profile still exists for the electron population. This density region is of one order of magnitude lower than that of the 30 cm case, an opposite of what was thought was going to happen. The ion population throughout the simulation can be seen in figure 3.13. It takes longer for the ion population to get to converge than that of the 30 cm case. The ion distribution plots are shown in figure 3.14. This ion density contour plot is very similar to the 30 cm case, but with the highest density only around half of it, which is because the electron density is also less than that of the 30 cm case.

3.3.3 25 cm case

As expected, the magnetic field strength is still stronger for the 25 cm case with respect to the 30 cm case, shown in figure 3.15. The resulting electron distribution is shown in figure 3.16. The density contour is similar to that of the 30 cm case, but with one unexpected difference. The maximum density for this case is 2.3×10^{12} , which is higher than the 30 cm case. It is expected to be less than the 30 cm case following a trend set by the 20 cm case. As shown in the figure, the magnetic field does a better job, compare to the 30 cm case, to contain the electrons more within the center of the chamber. Therefore, the density there is higher. The ion population throughout the simulation can be seen in figure 3.17. The time to population convergence is closer to the 30 cm case than that of the 20 cm case. Resulting ion distribution is shown in figure 3.18. Ion density, since electron density is higher than that of the 30 cm case, is higher.

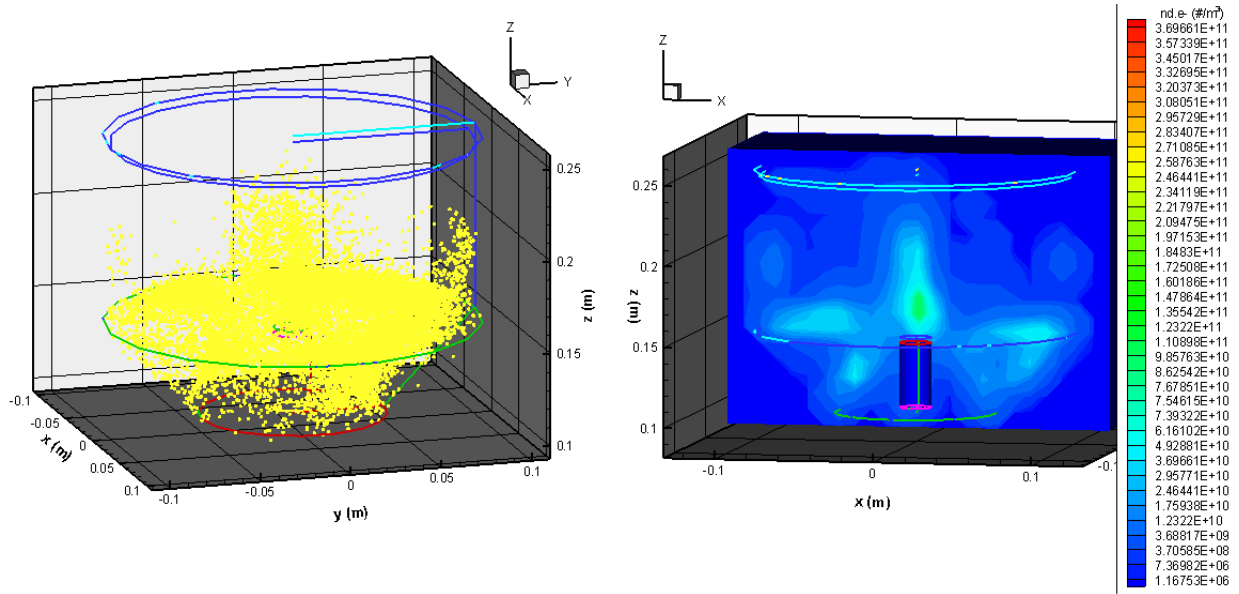


Figure 3.12: Electron distribution plots for the 20 cm thruster case after 1000 time steps. There still exists the densely populated region in the middle of the chamber, but with maximum density lower than that of the 30 cm case.

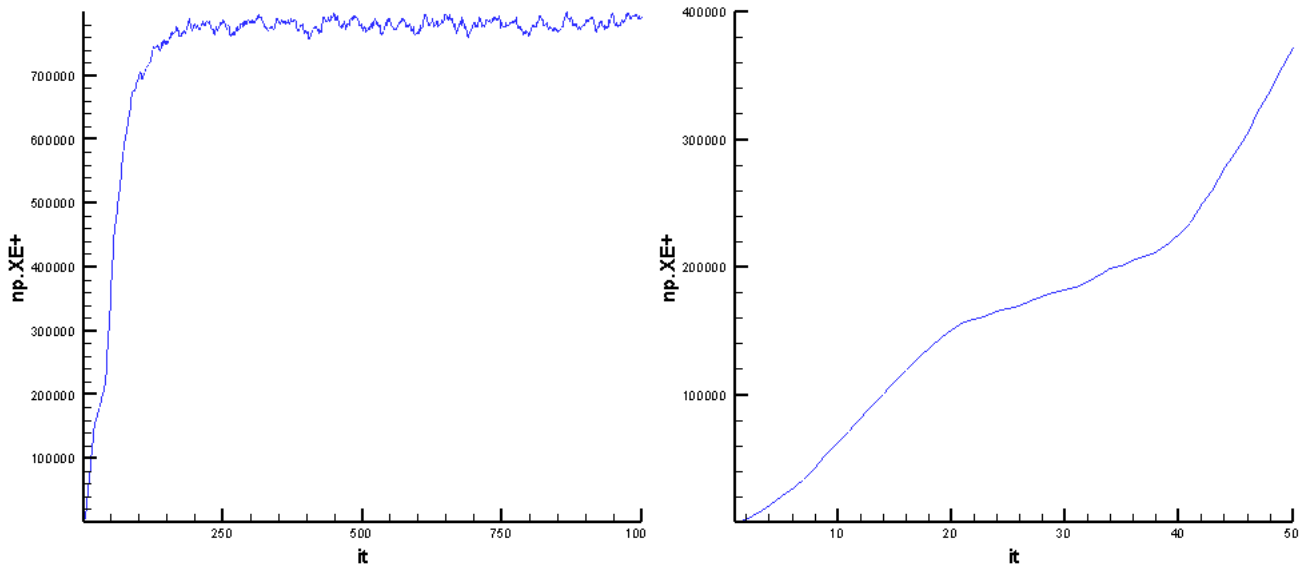


Figure 3.13: Ion population for the 20 cm case. The right plot shows the initial ionization for the first 50 time steps. The ionization rate is 7.735×10^{12} ions per second.

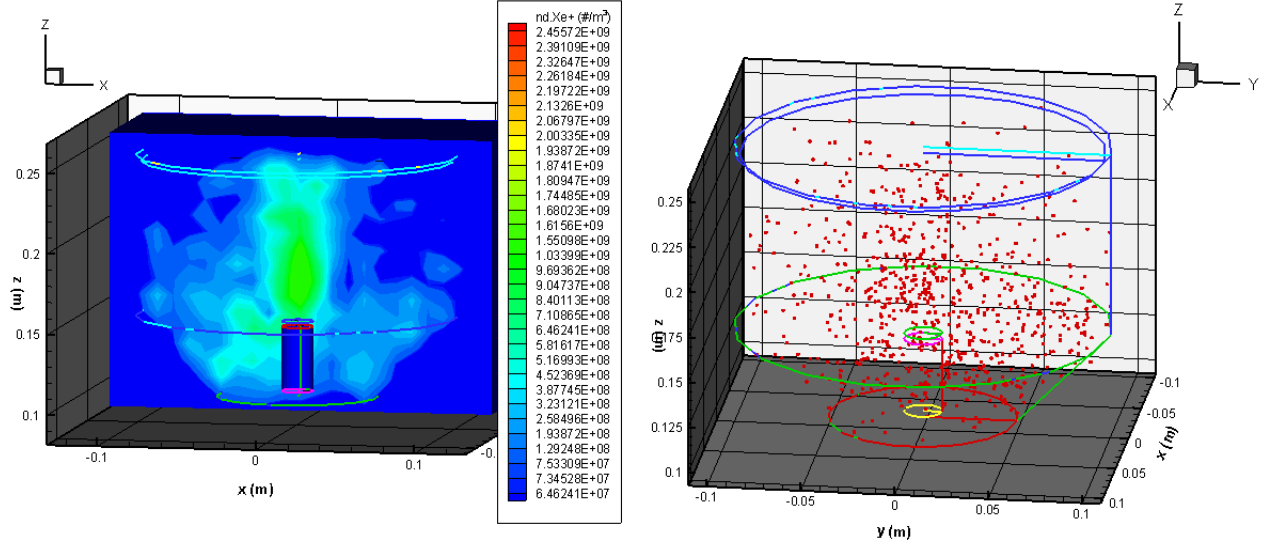


Figure 3.14: Ion distribution plots for the 20 cm thruster case after 1000 time steps. The maximum density for the high density region in the middle of the chamber is about half that of the maximum density of the 30 cm thruster case.

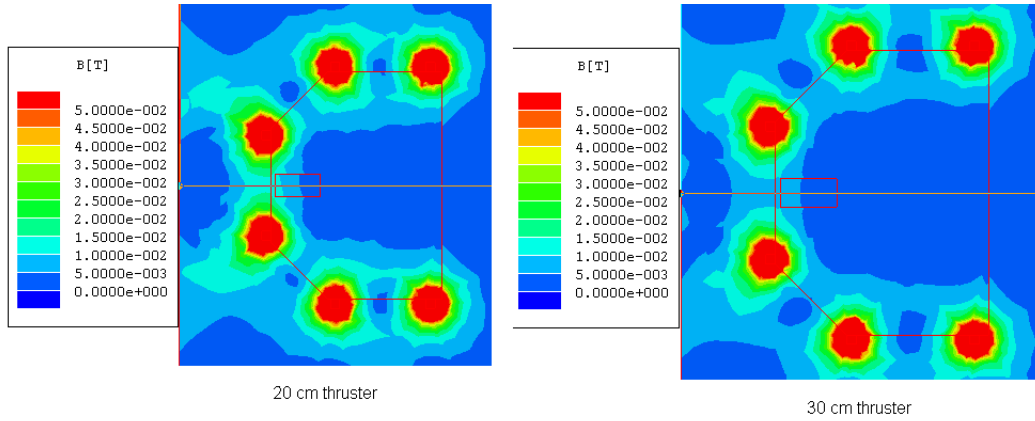


Figure 3.15: Magnetic field strength contour plots for the 25 cm and 30 cm case. The 25 cm case has a stronger magnetic field.

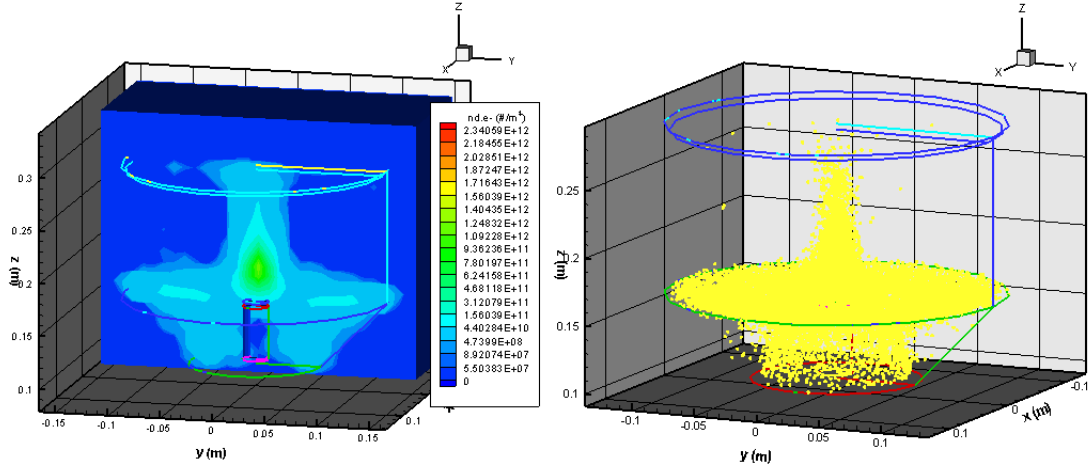


Figure 3.16: Electron distribution for the 25 cm thruster case. The maximum density at the center of the chamber is $2.3\text{e}12$, which is about double that of the 30 cm case.

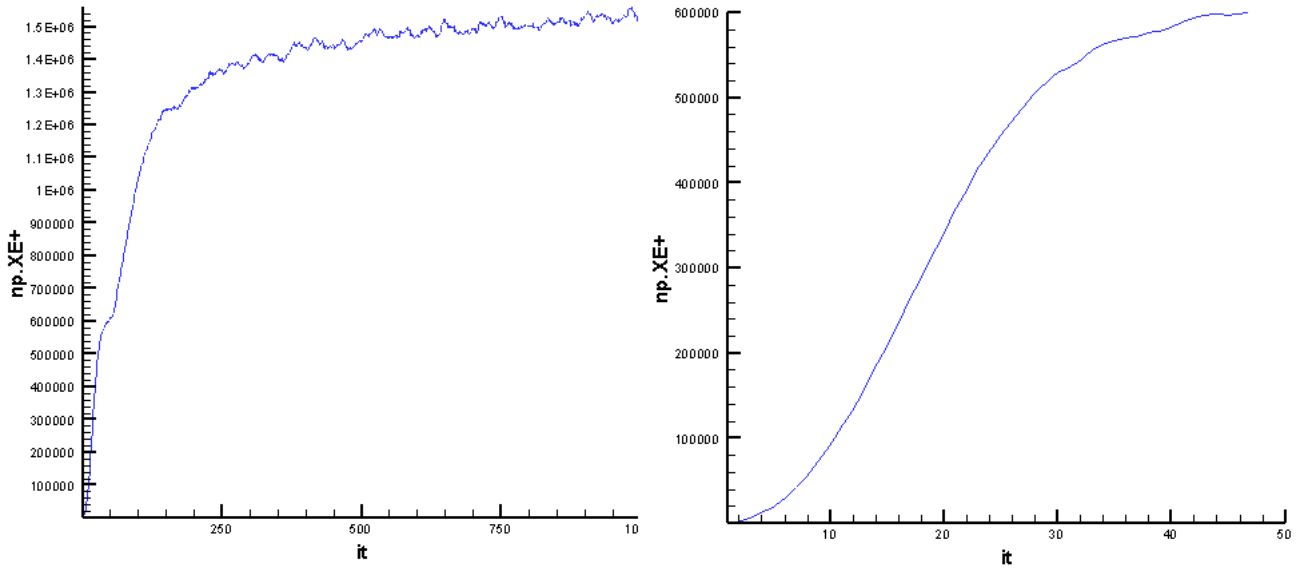


Figure 3.17: Ion population for the 25 cm case. The right plot shows the initial ionization for the first 50 time steps. The ionization rate is 1.295×10^{13} ions per second.

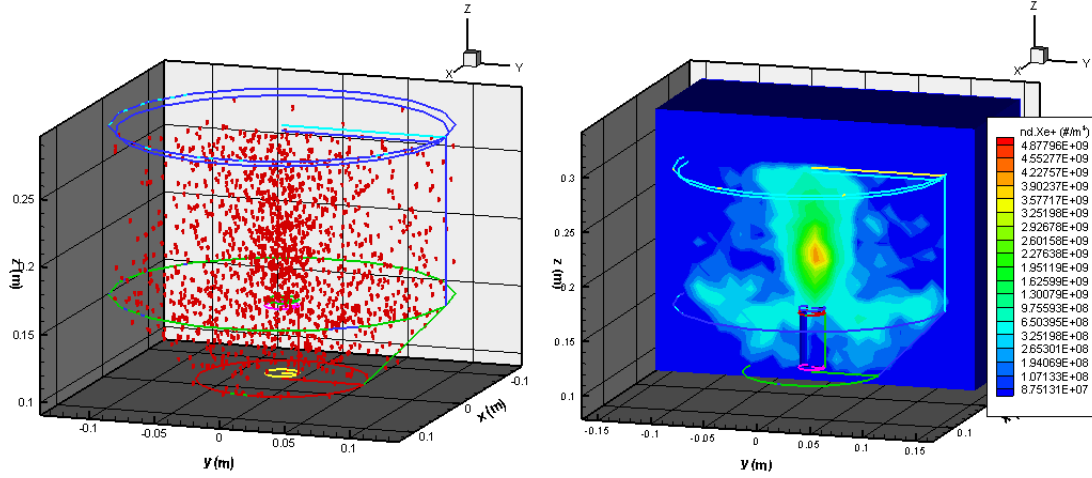


Figure 3.18: Ion distribution for the 25 cm thruster case. As expected, since electron density is higher, the ion density is higher than that of the 30 cm case.

3.3.4 35 cm case

Magnetic field of the 35 cm and the 30 cm case is shown in figure 3.19. The resulting electron distribution after 1000 time steps simulation run is shown in figure 3.20. The dense area in the center of the chamber appears to be larger than that of the 30 cm case, even though the maximum density is around half. This is because of the wider low field region in the middle allowing electrons to be more evenly distributed in this region. Ion population throughout the simulation can be seen in figure 3.21. Ion distribution is shown in figure 3.22. Ion density is lower than the 30 cm case, but the high density region in the center of the chamber is larger.

3.3.5 40 cm case

Magnetic field of the 40 cm and 30 cm case is shown in figure 3.23. The electron density is shown in figure 3.24. Electron density is less than the 30 and 35 cm case, but the high density region in the middle of the chamber is larger than both, same reasoning as previous. Ion population throughout the simulation can be seen in figure 3.25. Ion distribution is shown in figure 3.26. Density still follows the trend of being less than the 30 and 35 cm cases, but the high density region in the middle of the chamber is larger than both the other two cases. This concludes the analysis of particle population from simulation cases.

To look at the behavior of ionization rate between different chamber sizes, the ionization rate of each case is calculated from data from the 10th

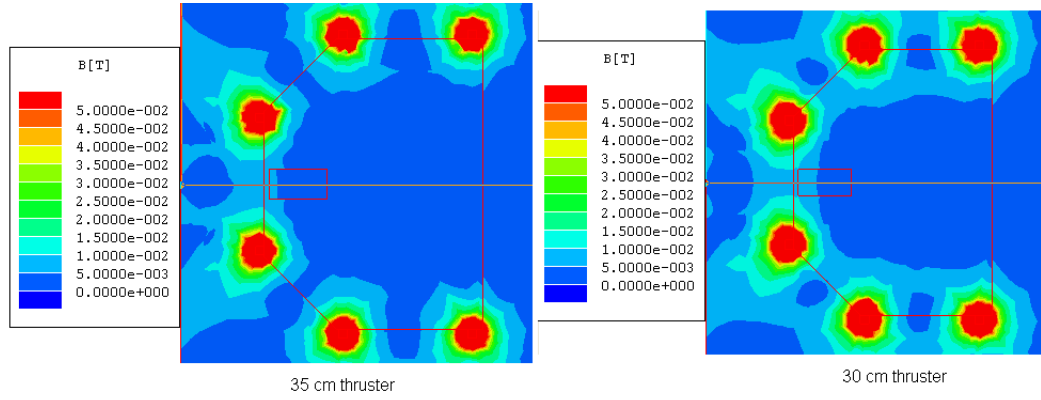


Figure 3.19: Magnetic field contour plots for 35 cm and 30 cm cases. Field strength for the 35 cm case is weaker than that of the 30 cm case.

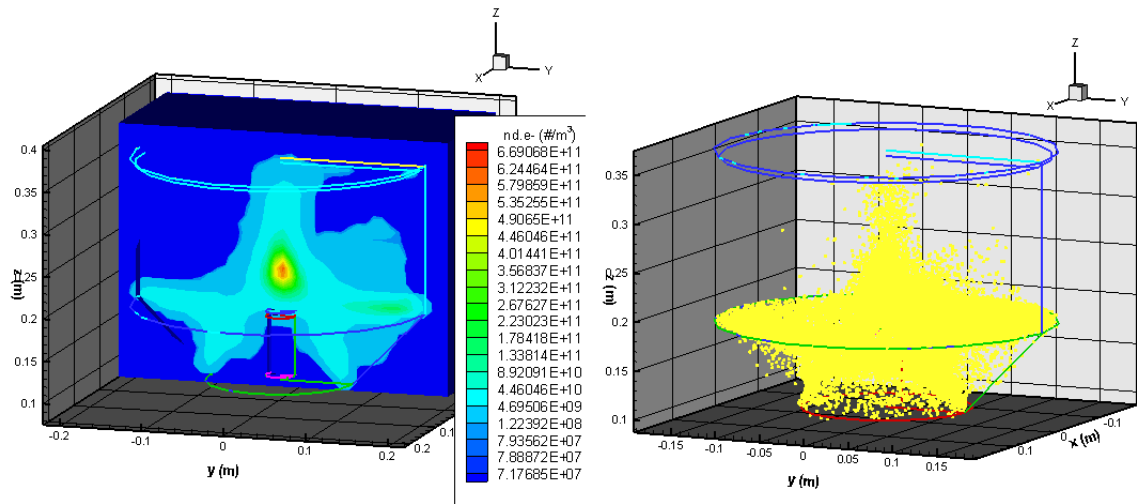


Figure 3.20: Electron distribution for the 35 cm case. Maximum density is around half of the 30 cm case.

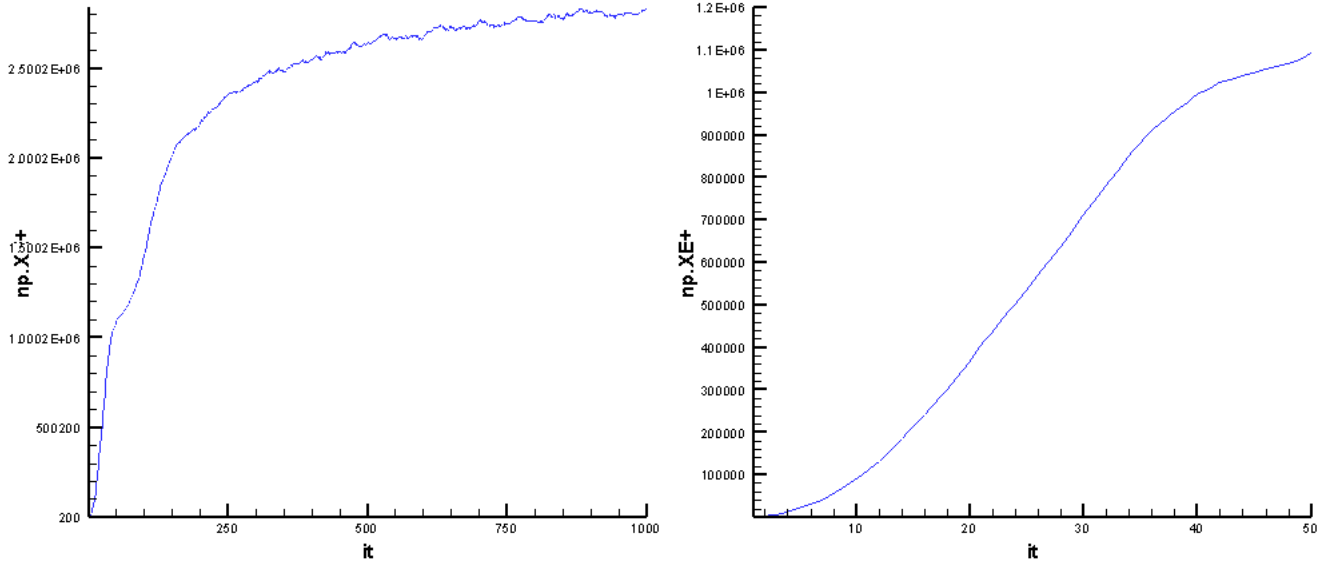


Figure 3.21: Ion population for the 35 cm case. The right plot shows the initial ionization for the first 50 time steps. The ionization rate is 2.505×10^{13} ions per second.

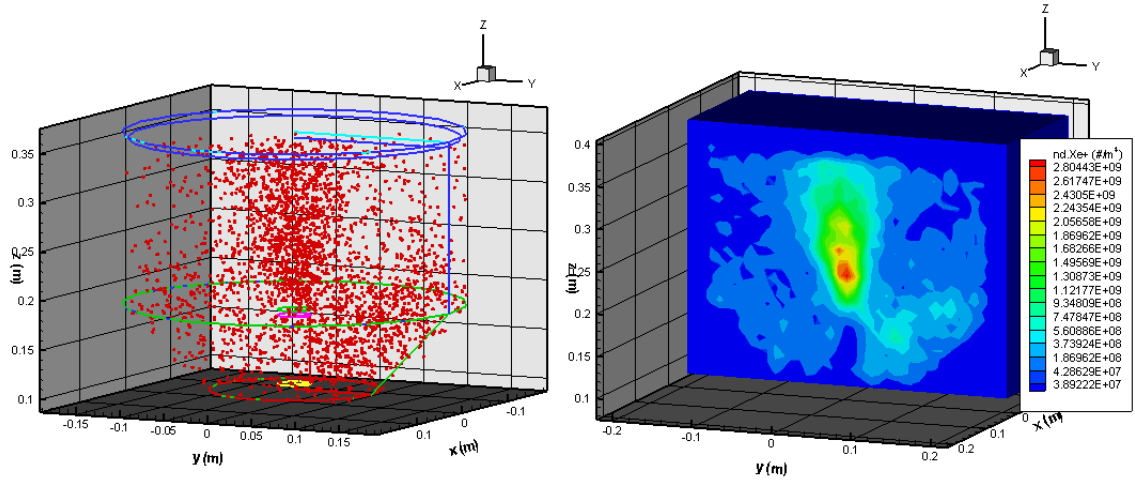


Figure 3.22: Ion distribution for 35 cm case. Following the electron density, ion density is lower than that of the 30 cm case, but the high density region is larger.

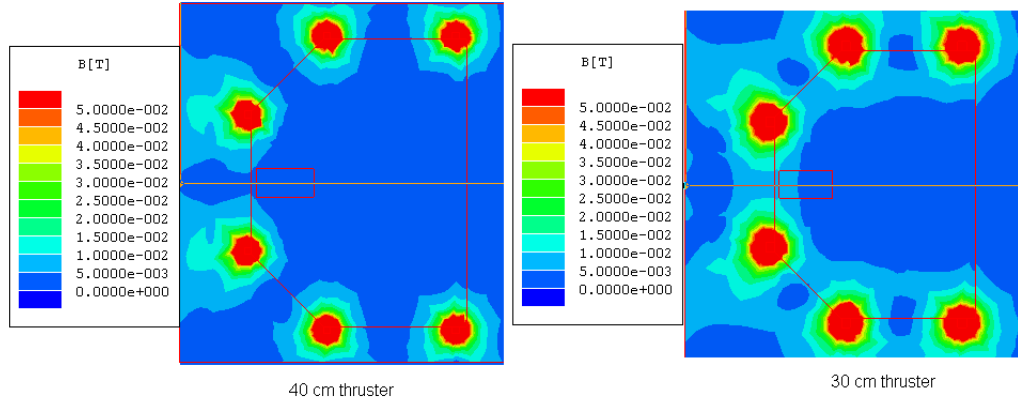


Figure 3.23: Magnetic field contour plots for 40 cm and 30 cm cases. Field strength for the 40 cm case is weaker than that of the 30 cm case.

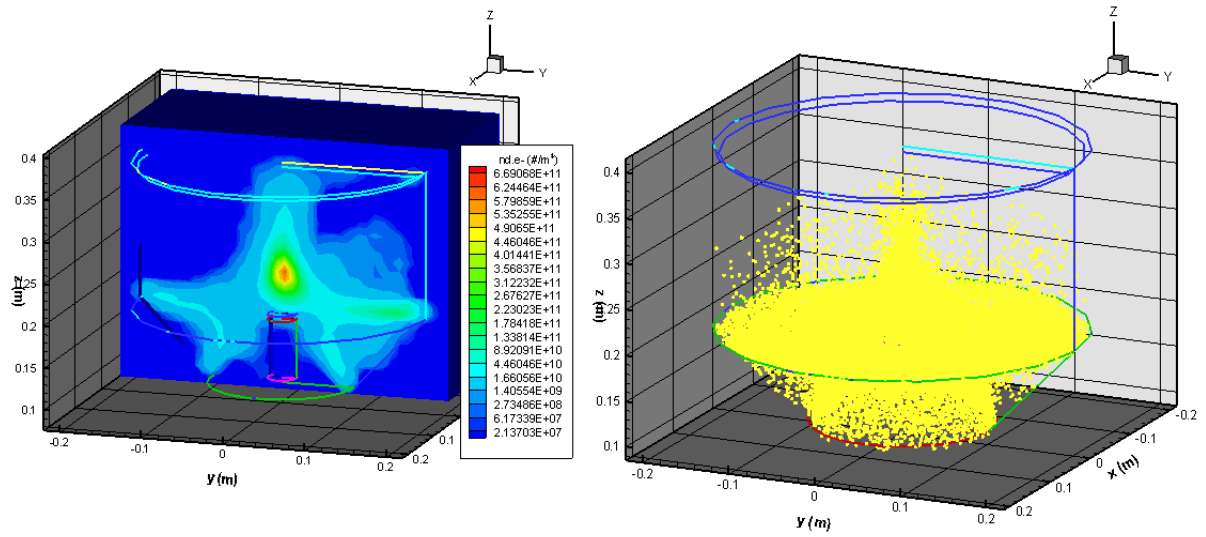


Figure 3.24: Electron distribution for the 40 cm thruster case. High density region in the middle is larger than that of the 30 and 35 cm cases.

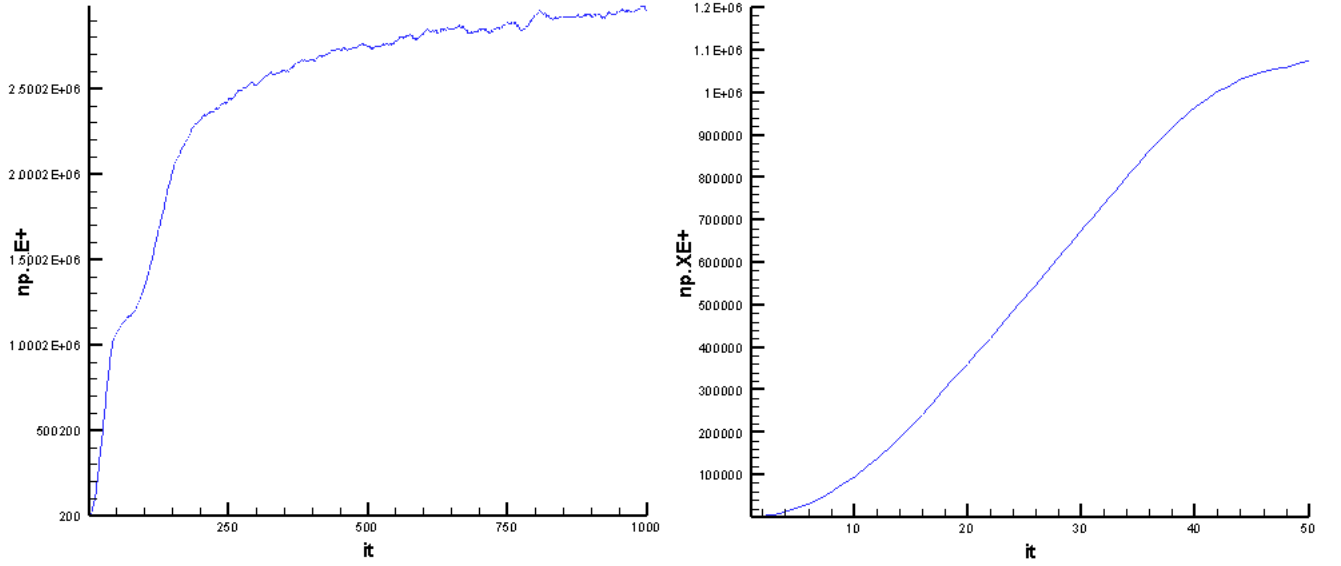


Figure 3.25: Ion population for the 40 cm case. The right plot shows the initial ionization for the first 50 time steps. The ionization rate is 2.444×10^{13} ions per second.

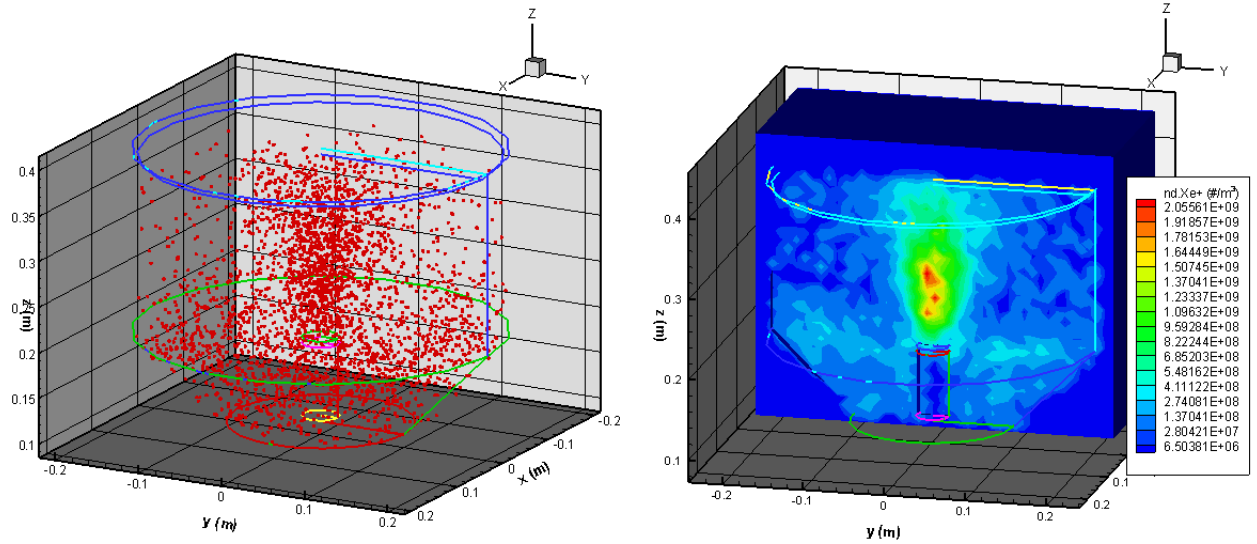


Figure 3.26: Ion distribution for the 40 cm thruster case. Density continues to follow the trend of being less, but with a wider region.

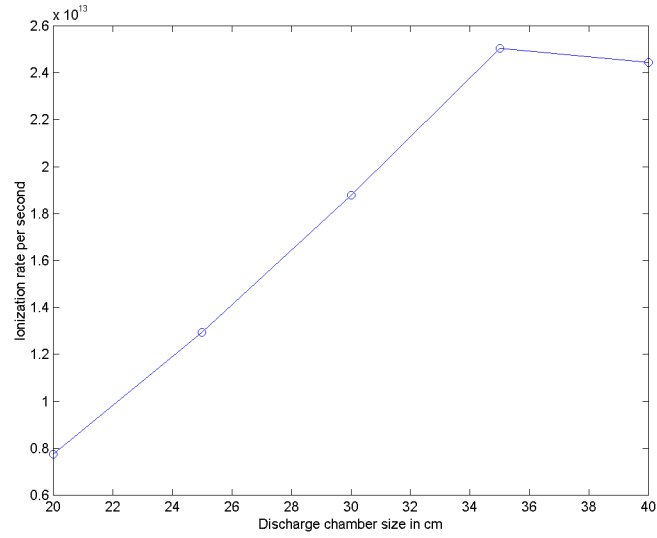


Figure 3.27: Ionization rate for the discharge chambers from the simulations with values in ions per second. The calculated ionization rate is 9×10^{14} ions per second.

and 50th time step. This data is then plotted and shown in figure 3.27. The plot shows that the optimum rate is in fact not the 30 cm case, but the 35 cm case.

Chapter 4

Application in the Performance model

4.1 Performance curve development

All the simulation cases are analyzed and appear to be correct. The next step now is to use those output data to develop a performance curve. To do this, the flux values from the simulations output data are used to define parameters in the Brophy's model to calculate corresponding energy cost values. Due to the limitation of the current state of the simulation setup, a few parameters will have to be assumed, mainly the primary electron utilization factor C_0 , the propellant flow rate $\dot{m}$, the propellant utilization efficiency η_u , and the baseline plasma energy cost ϵ_P^* . Since the number density of the xenon is assumed to be significantly larger than that of the other two species, and therefore assumed to be constant, the simulation does not actually inject xenon into the domain. Rather, the number density of xenon is hard coded into the program and use as a constant. Since there is no actual propellant flow rate, a value is adopted from [2] to be 1 equivalent amp. The other parameter, which is also taken from the same source, is the baseline plasma ion energy cost, ϵ_P^* , to be 50 eV. The propellant utilization efficiency has to be assumed since it is based on the rate of ionization over the rate of neutral injection. The definition of the primary electron utilization is defined as

$$C_0 = \frac{4\sigma_0 l_e}{ev_0 A_g \phi_0} \quad (4.1)$$

The most important variable in this equation is the primary electron containment length l_e . At the current state, this simulation setup does not provide this length and so the the primary electron utilization cannot be calculated. Therefore, the propellant utilization efficiency and the primary electron utilization are taken from [2] to be 0.1 and 3, respectively. The justification for assuming all these variables is that this thesis is aiming to

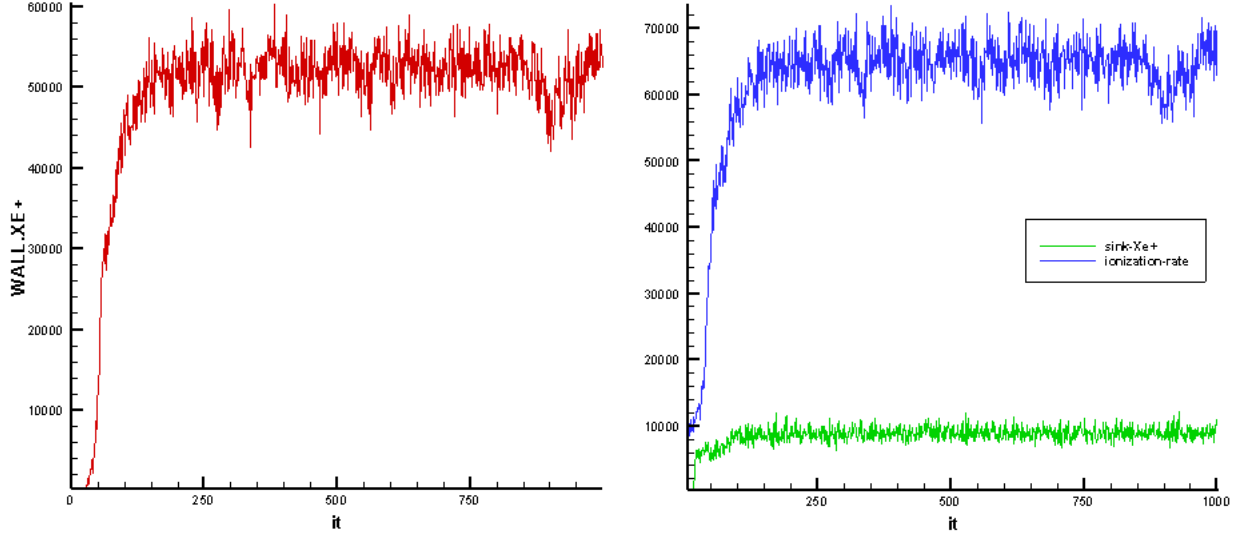


Figure 4.1: Ion flux to chamber wall and grid for the 20 cm thruster simulation case.

develop a relative performance model along differently sized thruster chambers. In talking about performance of a thruster, the most important aspect is how much of the ions in the chamber the thruster uses for thrust creation and how much is lost to the chamber wall. This aspect is contained in the two fractions, extracted-ion f_B and cathode surface-ion fraction f_C . Therefore, these two fractions will be calculated for each of the simulation cases and used in the energy equation 1.32 to create this performance model.

Ion flux to chamber wall and grid plots for each of the simulation cases are shown in figures 4.1 - 4.5. For each case, the average number of ions that are lost to the chamber and grid is determined and put in table 4.1. This table does not include ion flux to the surfaces of the discharge cathode assembly since it is only in the order of magnitude of a hundred per time step, insignificant relative to other values and therefore absorbed by them. The total ions produced is the average number of ions created per time step. Sink column represents the average number of ions collected by the sink surface per time step. The wall column is the average number of ions hitting the wall per time step. The beam ion loss column is only 70% of the sink ion loss to take into account the grid transparency. The cathode surfaces ion loss rate is the wall ion loss rate plus 30% of the sink loss rate. With this data, the extracted-ion and cathode surfaces-ion fractions are calculated. Also, the average beam ion energy cost for each of the cases is calculated. All of these values are shown in table 4.2 and the

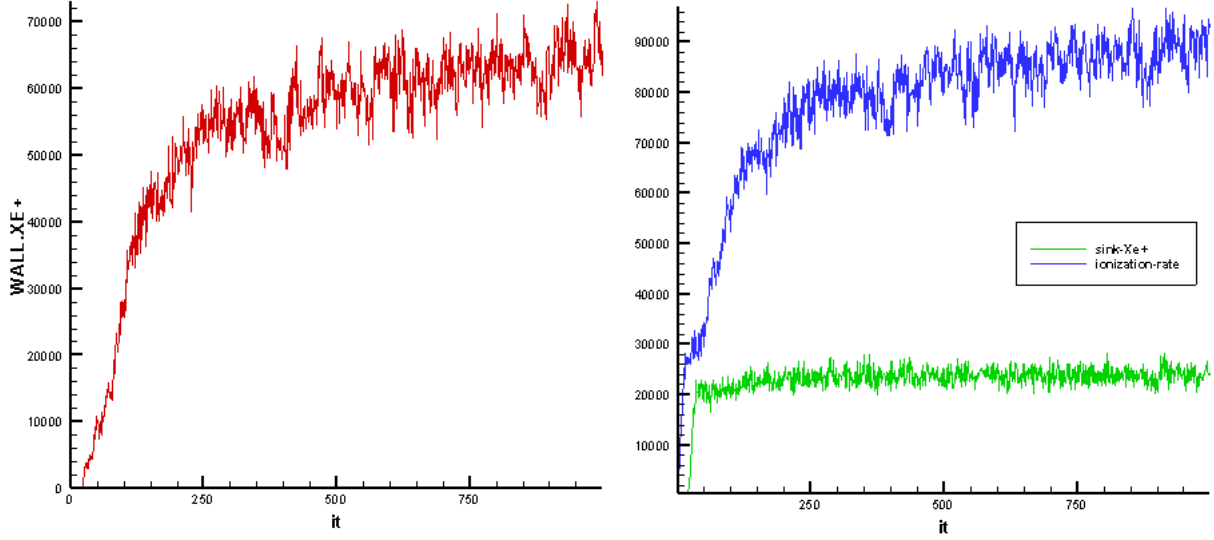


Figure 4.2: Ion flux to chamber wall and grid for the 25 cm thruster simulation case.

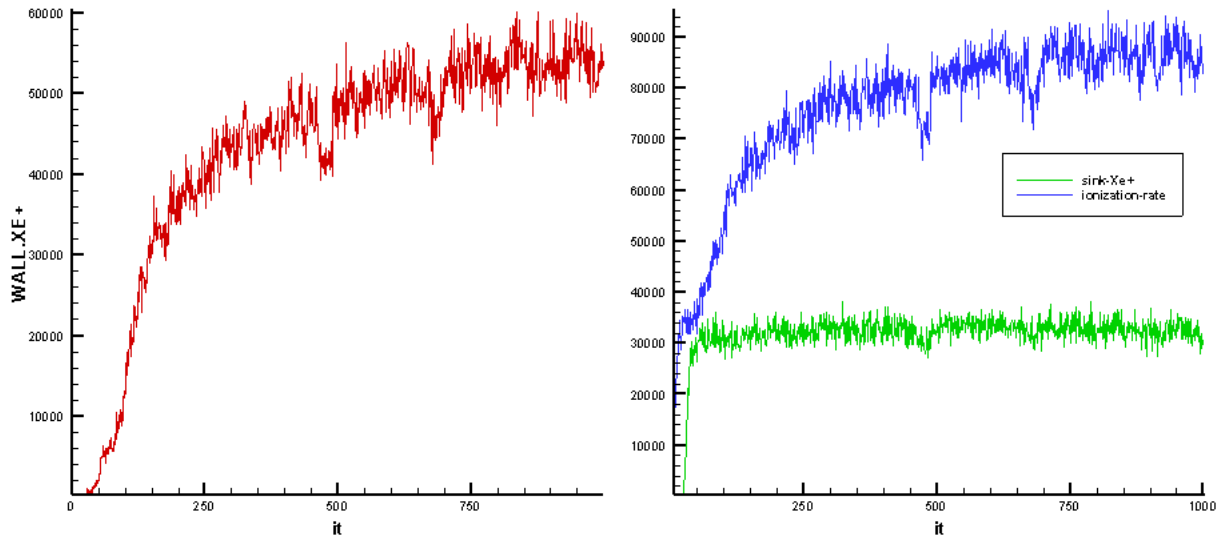


Figure 4.3: Ion flux to chamber wall and grid for the 30 cm thruster simulation case.

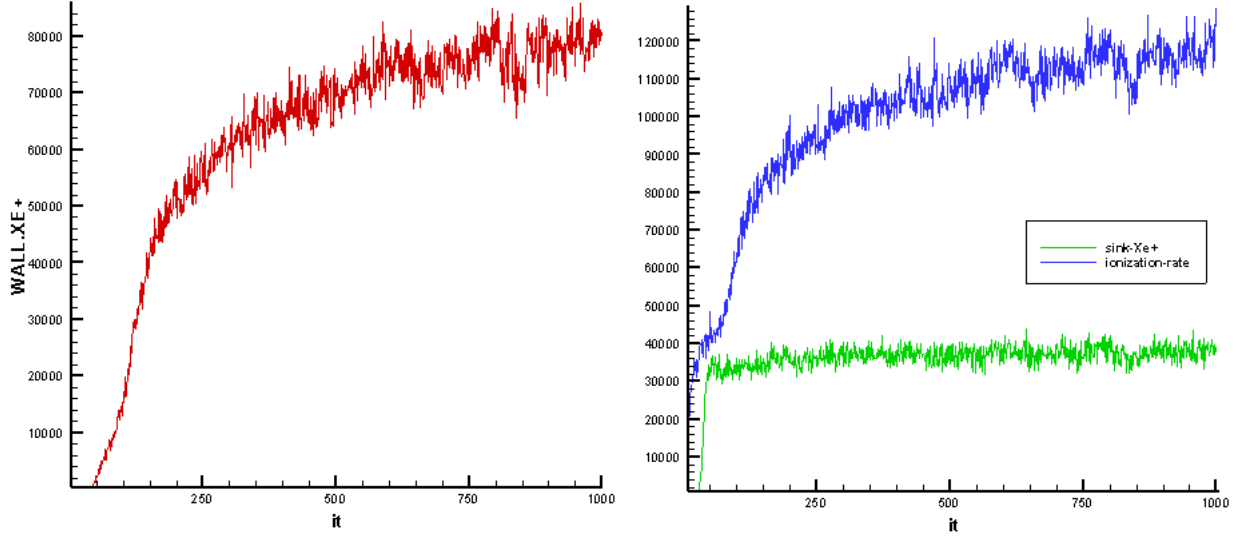


Figure 4.4: Ion flux to chamber wall and grid for the 35 cm thruster simulation case.

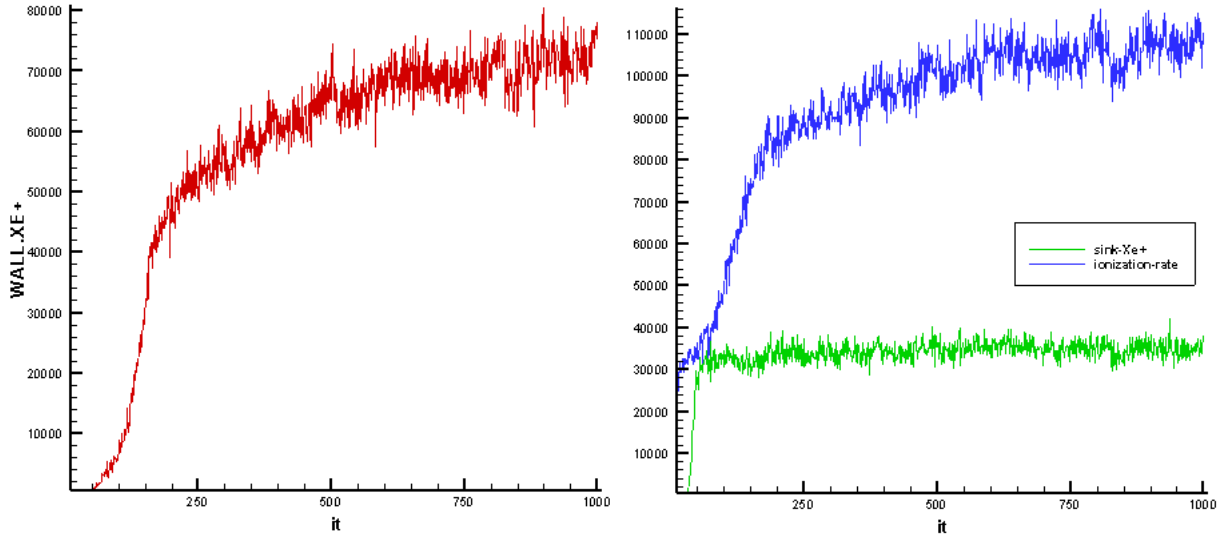


Figure 4.5: Ion flux to chamber wall and grid for the 40 cm thruster simulation case.

Thruster size	Total ions produced	Sink	Wall	Beam	Cathode surfaces
20 cm	63,000	9,000	54,000	6,300	8,100
25 cm	88,000	21,000	67,000	14,700	73,300
30 cm	86,000	31,000	55,000	21,700	64,300
35 cm	116,000	37,000	79,000	25,900	90,100
40 cm	105,000	33,000	72,000	23,100	81,900

Table 4.1: Flux data toward surfaces in ions per time step. The first three columns represent the raw data from the simulation. The fourth column is the number of ions going out of the thruster per time step, considering grid transparency of 0.7 and one time step being $1e^{-9}$ seconds. The last column is the number of ions lost to thruster cathode surfaces.

Thruster size	f_B	f_C	ϵ_B (eV)
20 cm	0.1	0.9	761.02
25 cm	0.167	0.833	445.67
30 cm	0.252	0.748	286.91
35 cm	0.223	0.777	327.48
40 cm	0.220	0.78	332.28

Table 4.2: Extracted-ion flux, cathode surfaces-ion flux, and average beam ion energy cost.

corresponding plot is shown in figure 4.6. The plot shows that the best thruster size for the parameters used is the 30 cm case. This is true because the operating parameters used are obtained from a 30 cm thruster. If operating parameters are changed for each of the thruster size, the 30 cm case might not remain the most efficient case. The expected behavior for the energy cost is to cost less as the chamber size increases. This behavior is expected because there is a statement in [2] about the primary electron length. It states that the length is expanded as the size of the chamber expands, therefore making the thruster more efficient. This is true up to the 30 cm thruster case where energy cost would rise again. An explanation for this is that the parameters used in the simulations are taken from a 30 cm thruster experiment. Because of this, these parameters are optimized for the 30 cm thruster, making it the most efficient. Therefore, the simulation process is again validated. The most important thing learned from this thesis project is a method of simulating ion thrusters. An obvious application for this method is to optimize operating parameters before fabrication. This thesis project holds operating parameters and varies the chamber sizes. But for an optimization project, the engineer can hold the chamber size constant and vary different operating parameters to study their effects on the energy cost.

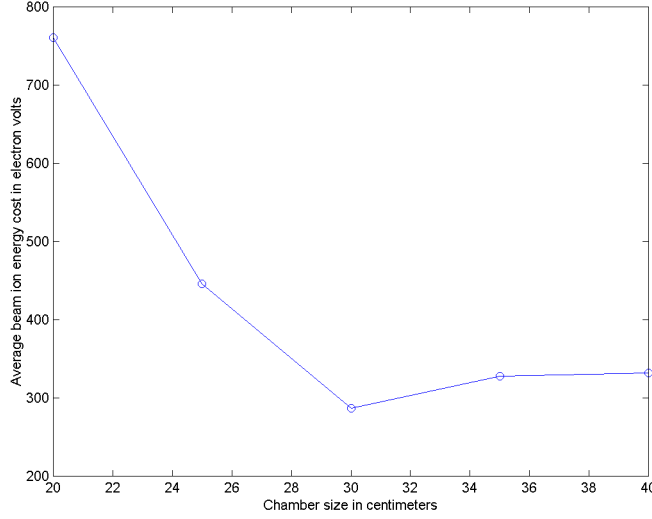


Figure 4.6: Average beam ion energy cost for five different chamber sizes with operating parameters constant. For this set of operating parameters, the best thruster size is the 30 cm case. This is not surprising since the parameters used are for a 30 cm thruster.

4.2 Electron loss behavior

In the previous section, only ion flux is considered for the performance model. This is due to the assumptions made, but another aspect of the thruster flux is also important. Electrons that are lost to surfaces also represent energy lost. Therefore this section will present the behavior of electron loss to determine whether there is a pattern as the chamber size changes. Plots of electrons lost to the chamber wall and grid are shown in figures 4.7 through 4.11 and the corresponding average flux values per time step is in table 4.3. The electron loss to wall and grid behaviors versus chamber size are shown in figure 4.12. In these two plots, the largest electron lost rate to the grid is in the 30 cm thruster case. Also in the 30 cm thruster case, there is the least electron flux toward the wall. The magnitude of the electron flux toward the chamber wall far exceeds that of the grid, figure 4.12 suggests that the 30 cm thruster case is the most efficient size for these operating parameters. It also suggests that there exists a force trapping the electrons and preventing them from moving into the grid surface. This containment efficiency is best looked through the ratio between the electron flux to the wall over the grid, shown in figure 4.13. Even though the 30 cm thruster is most efficient, in term of least amount of electron loss flux, the smallest thruster is most efficient at keeping electrons away from the grid. This is

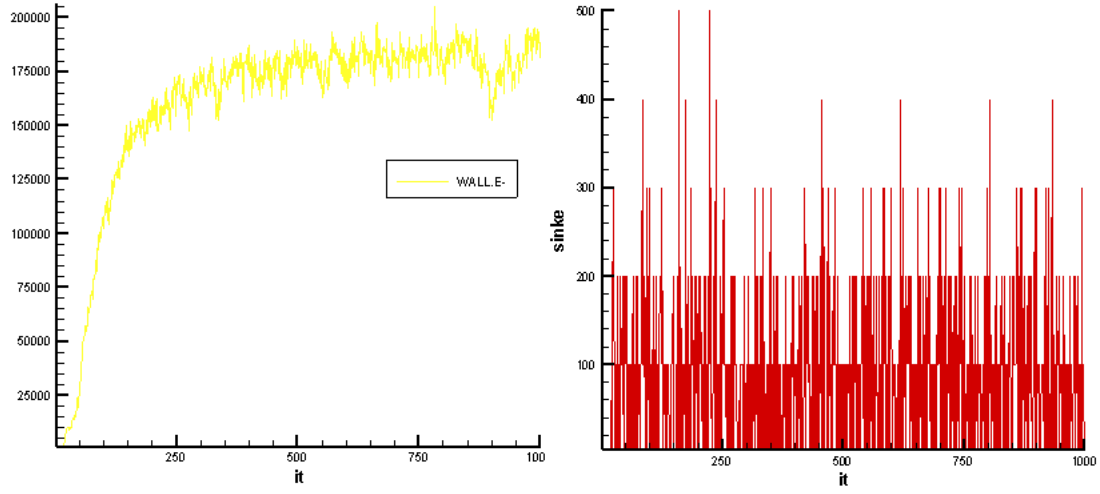


Figure 4.7: Electron lost plots for the 20 cm case. On the left is the flux toward the chamber wall and the right shows flux to the grid.

Thruster size	Flux toward wall	Flux toward grid
20 cm	180,000	100
25 cm	175,000	1,300
30 cm	145,000	1,500
35 cm	190,000	1,000
40 cm	160,000	400

Table 4.3: Average number of electrons loss to either the chamber wall or the grid in term of number per time step.

to be expected since the smallest thruster chamber size means the strongest presence of the magnetic field.

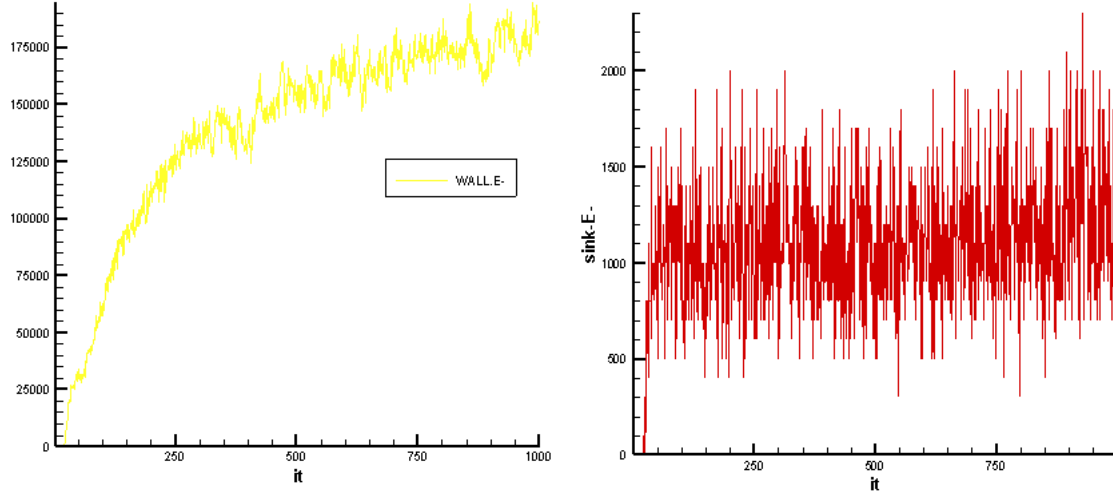


Figure 4.8: Electron lost plots for the 25 cm case. On the left is the flux toward the chamber wall and the right shows flux to the grid.

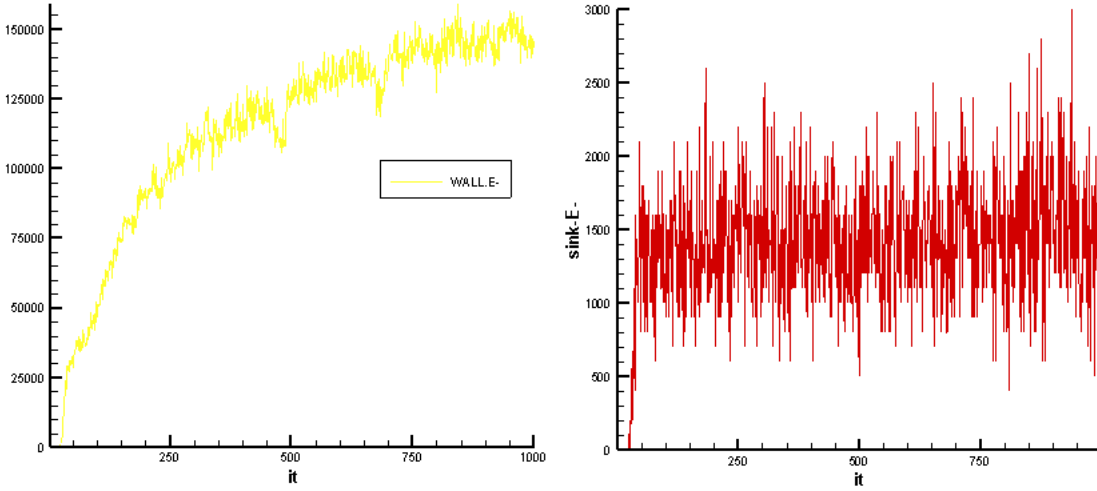


Figure 4.9: Electron lost plots for the 30 cm case. On the left is the flux toward the chamber wall and the right shows flux to the grid.

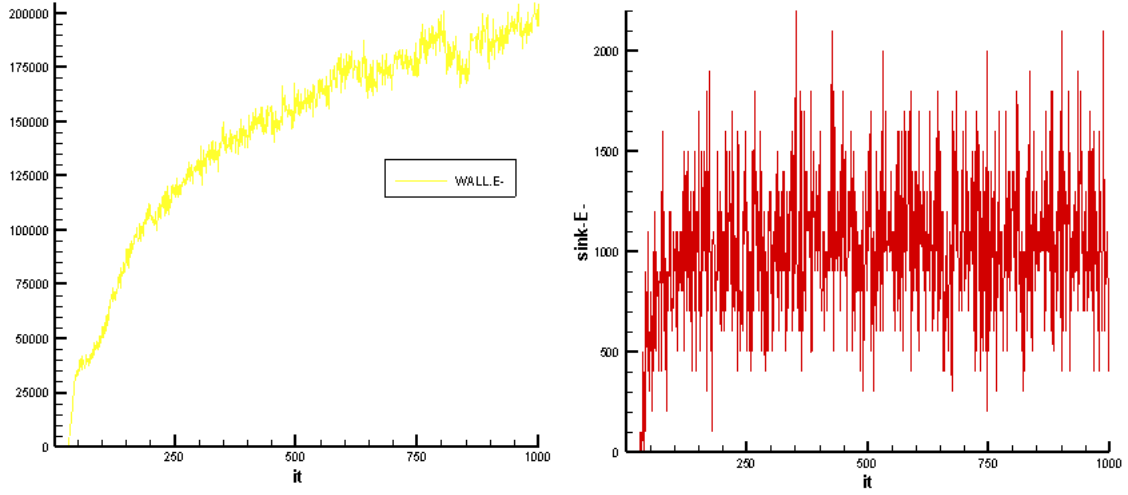


Figure 4.10: Electron lost plots for the 35 cm case. On the left is the flux toward the chamber wall and the right shows flux to the grid.

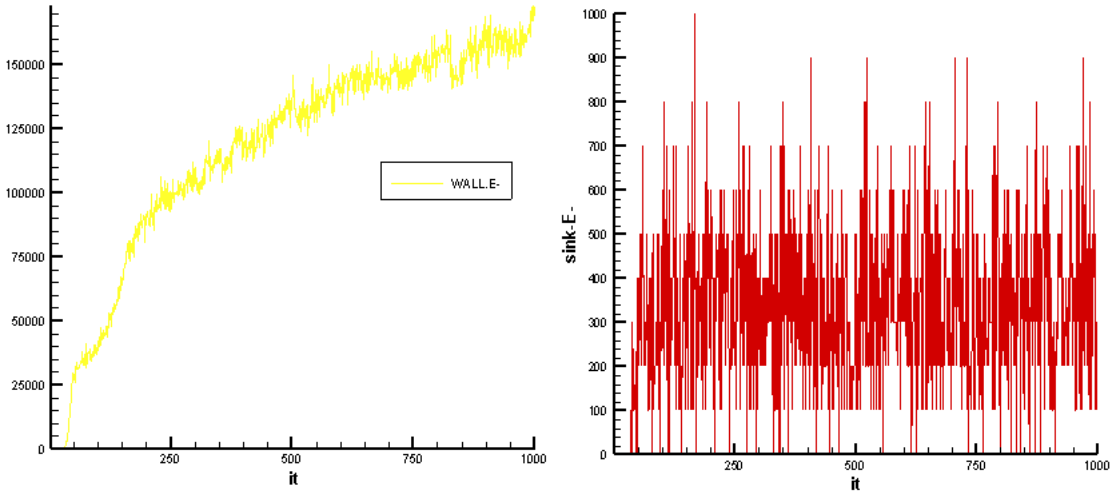


Figure 4.11: Electron lost plots for the 40 cm case. On the left is the flux toward the chamber wall and the right shows flux to the grid.

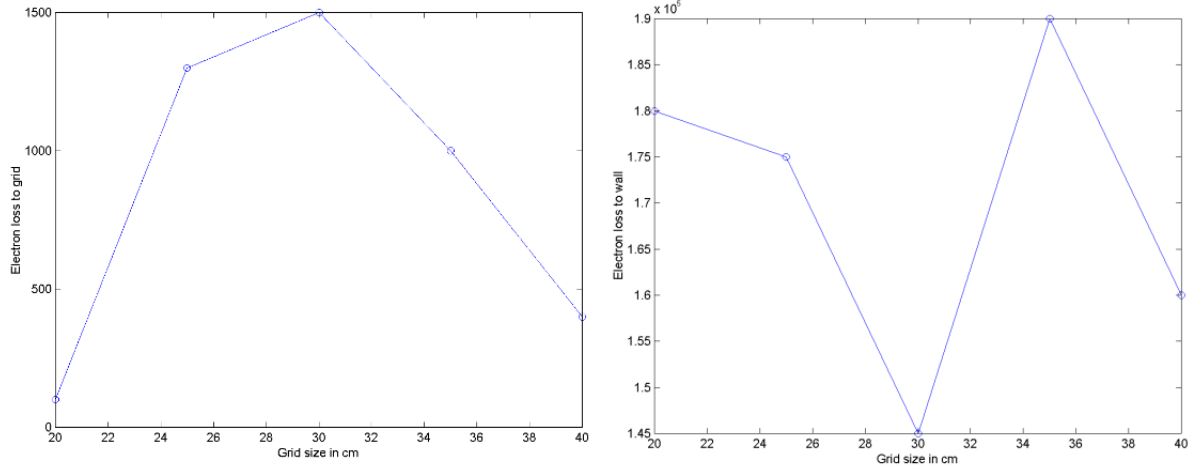


Figure 4.12: Electron loss to grid and wall behavior where the left plot shows a peak of electron loss to the grid in the case of the 30 cm thruster. The right plot shows the behavior of electron loss to the chamber wall with the least amount of loss per time step in the case of the 30 cm thruster.

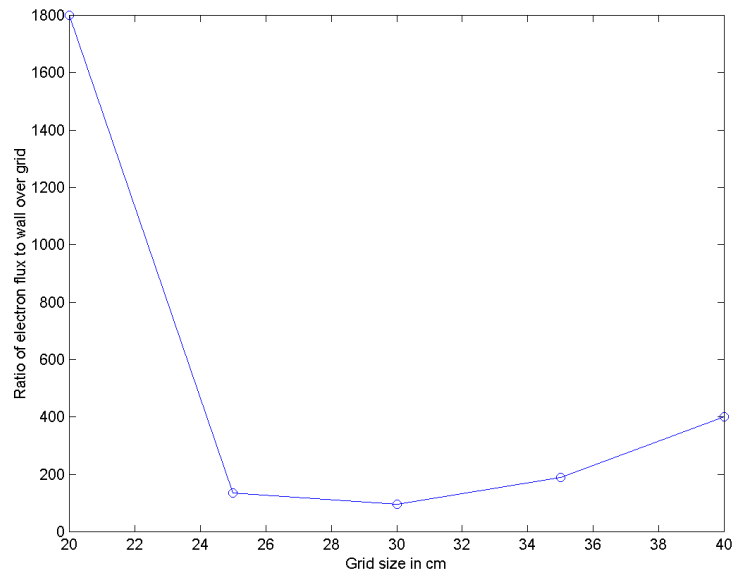


Figure 4.13: Ratio of electron flux to wall over grid

As a side study to see the effect of an additional magnet ring on the behavior of the electron population, a simulation with a fourth magnet ring placed in between the middle and one at the grid is conducted for 500 time steps with all other parameters the same as the 30 cm thruster simulation. This study only aims at the change of the electron distribution as a result of the extra magnet ring. The magnet placement and direction is shown in figure 4.14. As expected, electrons are also flowing toward the new magnet ring. Also, due to the ability to better contain electrons, the dense region in the middle of the chamber is larger than that of the 3 magnet rings configuration, but this does come with a cost. Since there is another ring cusp for electrons to flow to, more of them are lost to the wall. To prove this statement, the electron flux toward the wall between the three rings configuration is compared with that of the 4 rings configuration, shown in figure 4.16. The plots clearly show that while the average electron flux for the 3 magnet rings case at around 500 time steps is around 120,000 electrons per time step, the flux for the 4 magnet rings configuration is larger at around 160,000 electrons per time step. This comparison validates the statement that the more magnet rings a thruster has, the more electrons will be lost to the chamber wall. It would be interesting to see how the behavior of the electron flux toward the grid changes with the changing of the number of magnets. This is shown in figure 4.17. For the 4 rings configuration, it does a better job at keeping the electrons in the chamber away from the grid with its loss at around 1,100 electrons per time step versus the 1,300 electrons lost to the grid per time step for the 3 rings configuration. However, this is not enough to justify the 40,000 more electrons lost to the wall therefore making the 3 magnets configuration a better way to go. After the 500 time steps run, the resulting electron distribution is shown in figure 4.15.

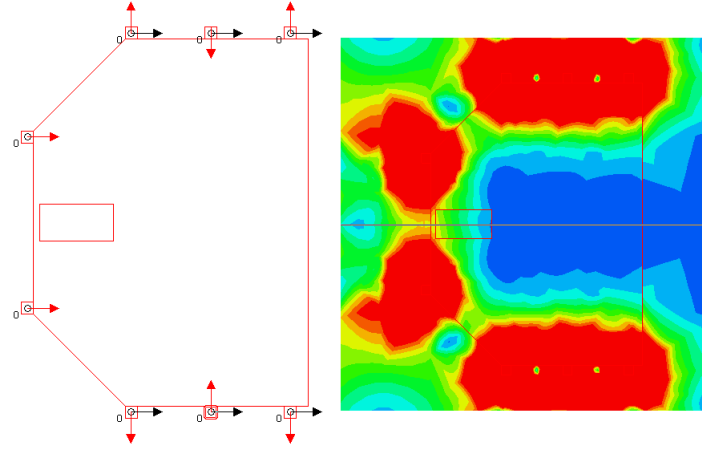


Figure 4.14: Magnet configuration and strength plot for 4 magnets. Field strength along the wall is stronger comparing to the 3 magnets case. Also, a larger area of low field strength in the middle of the chamber is present making electron containment better.

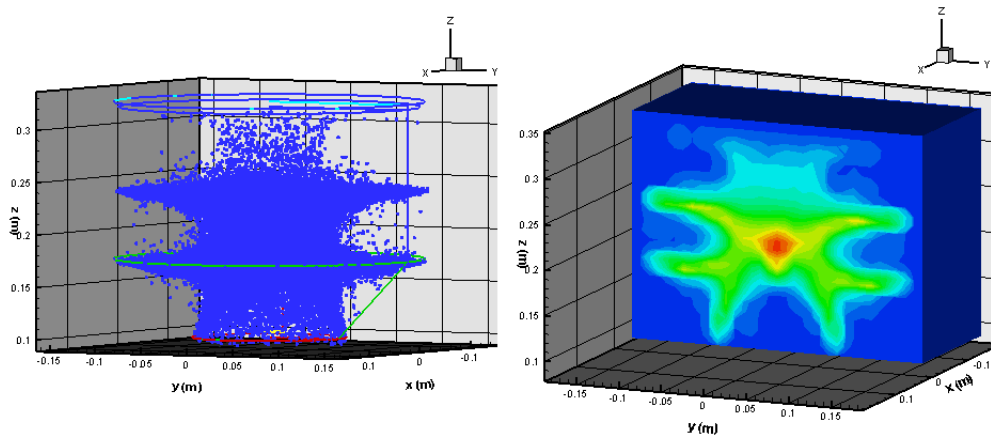


Figure 4.15: Electron distribution plots for the 30 cm thruster with 4 magnet rings. Electrons are group toward the cusps as expected.

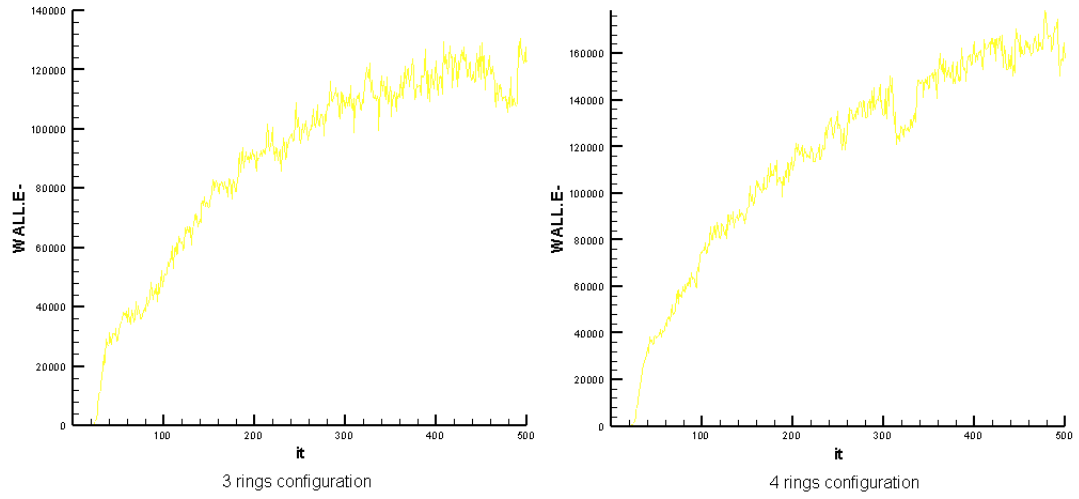


Figure 4.16: Electron flux to wall for 3 and 4 rings configuration

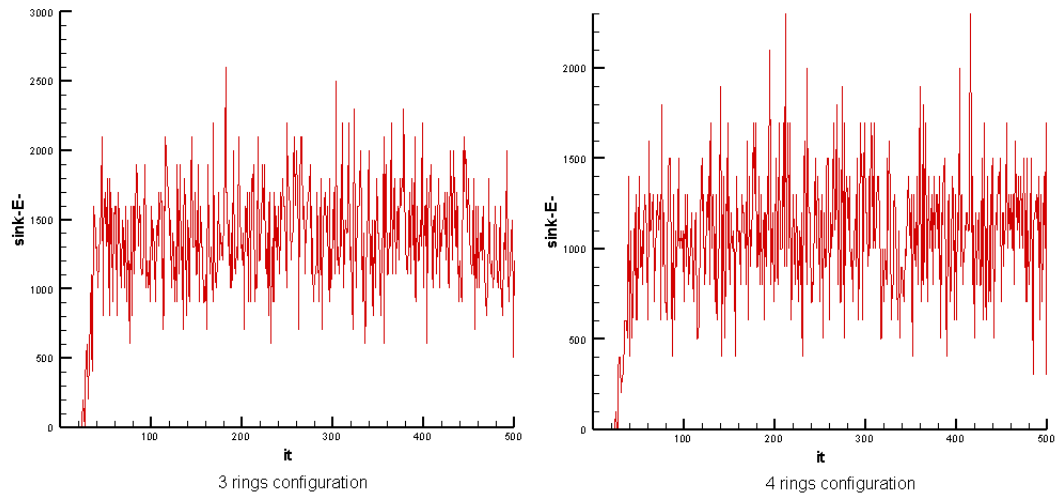


Figure 4.17: Electron flux to grid for 3 and 4 rings configuration

Chapter 5

Conclusions and Future Work

This thesis project was set out to develop a method to simulate ion thruster discharge chambers and it did just that. The process is completed using both commercially available and custom programs. Preliminary results from different cases are compared with other works and have been validated. As an application study, simulations of different thruster discharge chamber sizes are made to study the effect on the energy cost of creating beam ions. There are five different sizes of discharge chamber simulated while holding their operating parameter constant from thruster to thruster. The preliminary result shows that it costs more energy to generate beam ions if the size deviates from the original 30 cm thruster. This is because the operating parameters are taken from a 30 cm thruster. Therefore those parameters are already optimized for that size, further validating the simulation process. In this project, only one application case has been studied. There are numerous other ways to study the behavior of ion thrusters now that they can be simulated, thereby eliminating the restriction of cost from laboratory testings. To name a few

- Discharge Cathode Assembly erosion
- Full lifetime testing
- Effect of wall temperature on performance
- Effect of cathode mass flow rate on performance
- Different chamber configuration

Even though the simulation has been validated, it is still incomplete. Due to the lack of time and a complete model of the neutral density, the neutral density used is assumed constant throughout the chamber, but this is not the case. It actually decreases away from the cathode exit plane. Therefore,

it would make the process more complete if this neutral density distribution could be incorporated in the future. Also, the primary electron containment length is very important to the performance of the thruster itself. This model can simulate and track the trajectories of electrons. Therefore, it would be very beneficial to develop a method of calculating the average primary electron travel distance before ionization or loss. With this project as a base, hopefully many more applications can be thought up by other engineers to further complete this process.

Bibliography

- [1] Robert G. Jahn: *Physics of Electric Propulsion*, McGrawHill 1968
- [2] John R. Brophy and Paul J. Wilbur: *Simple Performance Model for Ring and Line Cusp Ion Thrusters*, AIAA Vol. 23, NO. 11, November 1985
- [3] John R. Brophy and Paul J. Wilbur: *An Experimental Investigation of Cusped Magnetic Field Discharge Chambers*, AIAA Vol. 24, NO.1, January 1986
- [4] Ronald W. Humble, Gary N. Henry, Wiley J. Larson: *Space Propulsion Analysis and Design*, McGrawHill 1995
- [5] <http://farside.ph.utexas.edu/teaching/plasma/lectures/node21.html>
- [6] <http://nmp.jpl.nasa.gov/ds1/>
- [7] Rodger J. Biasca *Numerical Simulation of the critical ionization velocity mechanism* PhD Dissertation, June 1992
- [8] Makoto Hayashi: *Determination of electron-xenon total excitation cross-sections, from threshold to 10 eV, from experimental values of Townsend's α* J. Physics. D: Appl. Phys., 16 (1983) 581-589. Printed in Great Britain
- [9] Daniel A. Herman and Alec D. Gallimore: *Comparison of Discharge Plasma Parameters in a 30cm NSTAR Type Ion Engine with and without Beam Extraction* AIAA paper
- [10] Matthew T. Domonkos and Alec D. Gallimore: *Parametric Investigation of Orifice Aspect-Ratio on Low Current Hollow Cathode Power Consumption* AIAA paper, 1998
- [11] Ira Katz, John R. Anderson, Dan M. Goebel, Richard Wirz, and Anita Sengupta: *Plasma Generation Near an Ion Thruster Discharge Chamber Hollow Cathode* AIAA, JPC July 2023 2003

- [12] James A. Menart and Michael J. Patterson: *Magnetic Circuit for Enhanced Discharge Chamber Performance of a Small Ion Thruster* AIAA 98-3343
- [13] Richard Wirz and Ira Katz: *2-D Discharge Chamber Model for Ion thrusters* AIAA, JPC 11 -14 July 2004

APPENDIX A

This appendix describes the process of converting output PATRAN files in neutral format into an input friendly file of TECPLOT format. The neutral file has two basic sections, top section contains information on the nodal positions and bottom section contains nodal connectivity information for elements. A sample of these two sections are shown in figure A-1 and A-2.

```
P3/PATRAN Neutral File from: /home/grad/bitran2/Thesis_project/Patran/dc30s.db
26      0      0      1      627      948      4      4      0
05-May-05 12:27:05      3.0
1      1      0      2      0      0      0      0      0
0.000000000E+0 0.000000000E+0 1.000000000E+1
1G      6      0      0 000000
1      2      0      2      0      0      0      0      0
0.000000000E+0 2.500000000E+0 1.000000000E+1
1G      6      0      0 000000
1      3      0      2      0      0      0      0      0
0.000000000E+0 5.000000000E+0 1.000000000E+1
1G      6      0      0 000000
1      4      0      2      0      0      0      0      0
0.000000000E+0 7.500000000E+0 1.000000000E+1
1G      6      0      0 000000
1      5      0      2      0      0      0      0      0
-8.550513983E-1 2.349231243E+0 1.000000000E+1
1G      6      0      0 000000
1      6      0      2      0      0      0      0      0
-1.710104823E+0 4.698462009E+0 1.000000000E+1
1G      6      0      0 000000
1      7      0      2      0      0      0      0      0
-2.565160036E+0 7.047691345E+0 1.000000000E+1
```

Figure A-1: Sample output for top part of a neutral PATRAN output file. The numbers 627 and 948 are number of nodes and cells, respectively. The number 4 after that is the number of different objects. After the line with date information, the data repeats in a 3 lines cycle. First line, second number is the node number. Second line contains the nodal position in x, y, and z coordinates. Third line and all other information in this 3 lines cycle are not important and therefore not used.

2	1	3	2	0	0	0	0	0
	3	0	1	0	0.000000000E+00	0.000000000E+00	0.000000000E+00	
	1	2	5					
2	2	3	2	0	0	0	0	0
	3	0	1	0	0.000000000E+00	0.000000000E+00	0.000000000E+00	
	2	3	5					
2	3	3	2	0	0	0	0	0
	3	0	1	0	0.000000000E+00	0.000000000E+00	0.000000000E+00	
	6	5	3					
2	4	3	2	0	0	0	0	0
	3	0	1	0	0.000000000E+00	0.000000000E+00	0.000000000E+00	
	7	6	3					
	3	0	2	0	0.000000000E+00	0.000000000E+00	0.000000000E+00	
	395	396	400					
2	590	3	2	0	0	0	0	0
	3	0	2	0	0.000000000E+00	0.000000000E+00	0.000000000E+00	
	401	400	396					
2	591	3	2	0	0	0	0	0
	3	0	2	0	0.000000000E+00	0.000000000E+00	0.000000000E+00	
	402	401	396					
2	592	3	2	0	0	0	0	0
	3	0	2	0	0.000000000E+00	0.000000000E+00	0.000000000E+00	
	396	397	402					

Figure A-2: Second half of neutral output file containing nodal connectivity information, also with 3 lines cycle behavior. First line, second number is the element number. Second line, third number is the object number. Third line contains nodal connectivity for the respective element. Elements are triangles therefore there are only three nodes assigned to each.

A custom C++ program picks out nodes and elements data and output them into a TECPLOT format file for input into the 3D PIC MCC code. This program (modified from a neutral to capVT program, from Lubos Brieda) is shown below.

```
#include<iostream>
#include<fstream>
#include<string>

using namespace std;

main()
{
float num;
unsigned int nodes,cells,node_number,element_number,num_mat;
unsigned int n1,n2,n3;
float x,y,z;
double lambda = 1e-2;//meters

ifstream in("dc30s.out");
ofstream out("dc30s.ans");
```

```

in.ignore(2000, '\n');
in.ignore(2000, '\n');

in>>num>>num>>num>>num;
in>>nodes>>cells>>num>>num_mat;
in.ignore(2000, '\n');
in.ignore(2000, '\n');

for( int ii=1;ii<=num_mat;ii++)
{
out<<"MP,EX,"<<ii<<","<<ii<<endl;
}

unsigned int index,*nf_ref,*nb_ref;

nf_ref=new unsigned int[nodes];

for (index=0;index<nodes;index++)
{
in>>num>>node_number;
in.ignore(2000, '\n');
in>>x>>y>>z;
in.ignore(2000, '\n');
in.ignore(2000, '\n');
in>>ws;
nf_ref[index]=node_number;
out<<"N, "<<node_number<<","<<x*lambda<<","<<y*lambda<<","<<z*lambda<<endl;
}

//out<<"NODES_END"<<endl;
unsigned int max=0;
for (index=0;index<nodes;index++)
if (nf_ref[index]>max) max=nf_ref[index];

nb_ref=new unsigned int[max];

for (index=0;index<nodes;index++)
nb_ref[nf_ref[index]]=index+1;

int mat_num = 0;
int temp;

for (index=0;index<cells;index++)

```

```

{
in>>num>>element_number;
in.ignore(2000,'\n');
in>>num>>num>>temp;
in.ignore(2000,'\n');
if( temp != mat_num )
{
mat_num = temp;
out<<"TYPE,"<<temp<<endl;
out<<"MAT,"<<temp<<endl;
}
in>>n1>>n2>>n3;
in.ignore(2000,'\n');
out<<"EN, "<<element_number<<", "<<n1<<", "<<n2<<", "<<n3<<endl;
}

return 0;
}

```

Samples of an output file are shown in figure A-3 and A-4

```

MP,EX,1,1
MP,EX,2,2
MP,EX,3,3
MP,EX,4,4
N, 1, 0, 0, 0.1
N, 2, 0, 0.025, 0.1
N, 3, 0, 0.05, 0.1
N, 4, 0, 0.075, 0.1
N, 5, -0.00855051, 0.0234923, 0.1
N, 6, -0.017101, 0.0469846, 0.1
N, 7, -0.0256516, 0.0704769, 0.1
N, 8, -0.0160697, 0.0191511, 0.1
N, 9, -0.0321394, 0.0383022, 0.1
N, 10, -0.0482092, 0.0574532, 0.1
N, 11, -0.0216507, 0.0125, 0.1
N, 12, -0.0433013, 0.0249999, 0.1
N, 13, -0.064952, 0.0374998, 0.1
N, 14, -0.0246202, 0.00434117, 0.1
N, 15, -0.0492404, 0.00868226, 0.1
N, 16, -0.0738606, 0.0130233, 0.1
N, 17, -0.0246202, -0.00434125, 0.1
N, 18, -0.0492404, -0.00868259, 0.1
N, 19, -0.0738605, -0.013024, 0.1

```

Figure A-3: Sample output of a TECPLOT format file. This top part contains nodal placement information with the first 4 lines indicating that there are 4 objects in this file. Every lines after these 4 contain node number then nodal coordinates.

```

TYPE,1
MAT,1
EN, 1, 1, 2, 5
EN, 2, 2, 3, 5
EN, 3, 6, 5, 3
EN, 4, 7, 6, 3
EN, 5, 3, 4, 7
EN, 6, 1, 5, 8
EN, 7, 9, 8, 5
EN, 8, 5, 6, 9
EN, 9, 6, 7, 9
EN, 10, 10, 9, 7
EN, 11, 1, 8, 11
EN, 12, 8, 9, 11
EN, 13, 12, 11, 9
EN, 14, 13, 12, 9
EN, 15, 9, 10, 13
EN, 16, 1, 11, 14
EN, 17, 15, 14, 11
EN, 18, 11, 12, 15
EN, 19, 12, 13, 15
EN, 20, 16, 15, 13

```

Figure A-4: Second part of file with the first line indicating object number. Lines after this contains element number and nodes that make up this element.

APPENDIX B

This appendix describes the process of converting a 2D output magnetic field distribution file from MaxwellSV into a 3D TECPLOT format magnetic field distribution file suitable for simulation input. An output file from MaxwellSV is in an R-Z coordinate system with magnetic field information in the respective axis. A sample part of such an output file is shown in figure B-1. The method to converting this field to a 3D field is relatively straight

```
|Vector data "Domain(ObjectList(-all-), <Br,Bz,0>)"
Grid Output Min: [0 -20 0] Max: [40 20 0] Grid Size: [1 1 1]
0 -20 0      0 0.0178800203653395 0
0 -19 0      0 0.0178800203653395 0
0 -18 0      0 0.0178800203653395 0
0 -17 0      0 0.0178800203653395 0
0 -16 0      0 0.0178800203653395 0
0 -15 0      0 0.0178800203653395 0
0 -14 0      0 0.0178800203653395 0
0 -13 0      0 0.0178800203653395 0
0 -12 0      0 0.0178800203653395 0
0 -11 0      0 0.0178800203653395 0
```

Figure B-1: A sample of an output 2D magnetic field file from MaxwellSV. Information is on a 2 dimensional R-Z plane with field strength in Tesla.

forward. In the R-Z plane, R is the horizontal axis and Z is the vertical axis. Therefore, R is chosen to be the Z axis in the 3D coordinate system. With this, the Z axis in the 2D coordinate system is rotated about its R axis to form a 3D cylindrical volume, making it into 3 dimensional. Since the 3 dimensional domain is a rectangular box and the rotated 2D information is a cylindrical, the gaps at the edges of the box needs to be filled in. This is taken care of by making the domain for the 2D field larger than the chamber making the edges not relevant to the simulation and therefore are set to zeroes. The whole conversion process has two steps. First step is to convert the 2D file into two files, one containing the coordinates and one containing the magnetic field information. The reason for this first step is that Matlab cannot, to the best knowledge of the author, take an input file with more than three columns of information. The second step is to use Matlab to make a 3D file out of the two files created by the first step. The first step program, in C++, is shown below. It is a very simple read in and output program.

```

#include<iostream>
#include<fstream>

using namespace std;

int main()
{

    ifstream in("bfield_2dvec_dc30s.dat");
    ofstream out1("B_xyz.dat");
    ofstream out2("B_hxyz.dat");

    int size;    //change size accordingly!!!
    double x,y,z,hx,hy,hz;

    cout<<"Input file length: "<<endl;
    cin>>size;

    //cout<<"number of rows: "<<size;

    for( int ii=0;ii<size;ii++ )
    {
        in>>x>>y>>z>>hx>>hy>>hz;
        in.ignore(2000,'\n');
        out1<<x<<" "<<y<<" "<<z<<" "<<endl;
        out2<<hx<<" "<<hy<<" "<<hz<<endl;
    }

    out1.close();
    out2.close();

    return 0;
}

```

The second step program, in Matlab, is shown below.

```

function temp = convertSC_vec(file, file2, dR, dZ)
temp=1;
R = file(:,1);
Z = file(:,2);
Hx = file2(:,2);
Hy = file2(:,2);
Hz = file2(:,1);

lambda = 1e-2; %in meters

```

```

factor = 1;

length_x = floor((max(Z) - min(Z))/dZ+1)
length_y = length_x;
length_z = floor((max(R)-min(R))/dR+1)
Htemp = 0;
origin_x = ceil(length_x/2)
origin_y = ceil(length_y/2)
origin_z = 0;

fid = fopen('Bfield_3dvec_dc30s.dat','w');
fprintf(fid,'%12s %6s %6s %6s %6s %6s %6s\n','VARIABLES = ', 'X', 'Y', 'Z', 'BFX', 'BFY', 'BFZ');
fprintf(fid,'%6s %4s %2f %4s %2f %4s %2f\n', 'ZONE', 'I= ',length_x, 'J= ',length_y, 'K= ',length_z);
R = R./dR; %convert double index to int index for comparision
R = round(R);
Z = Z./dZ;
Z = round(Z);
for z=min(R):max(R) %every point along axial direction
    indexr = find(R == z);
    Ztemp = Z(indexr);
    %Rtemp = R(indexr);
    Htempx = Hx(indexr); %only indices of z
    Htempy = Hy(indexr);
    Htempz = Hz(indexr);
    for y=min(Z):max(Z)
        for x=min(Z):max(Z)
            %-----
            r = (x^2 + y^2)^0.5;
            if (r > length_x/2-1)
                Hcx = 0;
                Hcy = 0;
                Hcz = 0;
            else
                %-----
            lower = find( Ztemp == floor(r) );
            upper = find( Ztemp == ceil(r) );
            if( upper == lower )
                lower = upper - 1;
            end

            Hcx = Htempx(lower)*(Ztemp(upper)-r) + Htempx(upper)*(r-Ztemp(lower));
            Hcy = Htempy(lower)*(Ztemp(upper)-r) + Htempy(upper)*(r-Ztemp(lower));
            Hcz = Htempz(lower)*(Ztemp(upper)-r) + Htempz(upper)*(r-Ztemp(lower));
            %Hcz = Htempz(lower);

```

```

%along axis lines
if( (x > 0) & (y == 0) )
    Hcy = 0;
end
if( (x < 0) & (y == 0) )
    Hcx = -Hcx;
    Hcy = 0;
end
if( (x == 0) & (y > 0) )
    Hcx = 0;
end
if( (x == 0) & (y < 0) )
    Hcy = -Hcy;
    Hcx = 0;
end
%quadrant I
if( (x>0) & (y>0) )
    theta = atan(abs(y)/abs(x));
    Hcx = Hcx*cos(theta);
    Hcy = Hcy*sin(theta);
end
%quadrant II
if( (x<0) & (y>0) )
    theta = atan(abs(y)/abs(x));
    Hcx = -Hcx*cos(theta);
    Hcy = Hcy*sin(theta);
end
%quadrant III
if( (x<0) & (y<0) )
    theta = atan(abs(y)/abs(x));
    Hcx = -Hcx*cos(theta);
    Hcy = -Hcy*sin(theta);
end
%quadrant IV
if( (x>0) & (y<0) )
    theta = atan(abs(y)/abs(x));
    Hcx = Hcx*cos(theta);
    Hcy = -Hcy*sin(theta);
end
end

fprintf(fid,'%6f %6f %6f %12.6f %12.6f %12.6f\n',x*dR*lambda,y*dR*lambda,
        z*dZ*lambda,Hcx*factor,Hcy*factor,Hcz*factor);
end

```

```

end

end
fclose(fid);

```

There is not really any strong explanation as to why use two separate languages for this process. It is just based on personal preference at the moment of coding. A sample of the output file is shown in figure B-2.

-0.140000	0.130000	0.080000	0.000627	-0.000582	-0.002640
-0.130000	0.130000	0.080000	0.001549	-0.001549	-0.002999
-0.120000	0.130000	0.080000	0.002193	-0.002375	-0.003205
-0.110000	0.130000	0.080000	0.002376	-0.002808	-0.003183
-0.100000	0.130000	0.080000	0.002500	-0.003250	-0.003162
-0.090000	0.130000	0.080000	0.002559	-0.003697	-0.003142
-0.080000	0.130000	0.080000	0.002549	-0.004142	-0.003124
-0.070000	0.130000	0.080000	0.002447	-0.004544	-0.003365
-0.060000	0.130000	0.080000	0.002257	-0.004891	-0.003839
-0.050000	0.130000	0.080000	0.002023	-0.005261	-0.004239
-0.040000	0.130000	0.080000	0.001778	-0.005778	-0.004531
-0.030000	0.130000	0.080000	0.001432	-0.006207	-0.004763
-0.020000	0.130000	0.080000	0.001004	-0.006529	-0.004932
-0.010000	0.130000	0.080000	0.000518	-0.006729	-0.005034

Figure B-2: Sample content of a 3D B field file. Values are now in XYZ and Bx By Bz with magnetic field strength in Tesla.